



ЕГЭ

Под редакцией
Л.Н. Евич, С.Ю. Кулабухова

ИНФОРМАТИКА и ИКТ

ПОДГОТОВКА
К **ЕГЭ-2015**

По новой демоверсии на 2015 год!

Пособие с электронным
приложением



(CD-диск)



УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС
«ИНФОРМАТИКА и ИКТ. ПОДГОТОВКА К ЕГЭ»

**Учебно-методический комплекс
«Информатика и ИКТ. Подготовка к ЕГЭ»**

Под редакцией Л. Н. Евич, С. Ю. Кулабухова

ИНФОРМАТИКА И ИКТ

ПОДГОТОВКА К ЕГЭ–2015

**Пособие с электронным
приложением (CD–диск)**

Учебно-методическое пособие



ЛЕГИОН
Ростов-на-Дону
2014

ББК 74.263.2

Е 16

Рецензент:

С. О. Иванов — руководитель отдела математики ООО «Легион»

Евич Л. Н.

Е16 Информатика и ИКТ. Подготовка к ЕГЭ-2015. Пособие с электронным приложением (CD-диск) / Под ред. Л. Н. Евич, С. Ю. Кулабухова — Ростов-на-Дону: Легион, 2014. — 272 с. — (Готовимся к ЕГЭ)

ISBN 978-5-9966-0619-1

Учебно-методическое пособие содержит необходимый материал и рекомендации для самостоятельной подготовки к ЕГЭ по информатике и ИКТ:

- 12 авторских учебно-тренировочных тестов, составленных по актуальным нормативным документам ЕГЭ (проект демоверсии и спецификации, опубликованные ФИПИ 30.08.2014) и с учётом опыта прошедшего экзамена;

- подробные решения заданий части 2;

- краткий теоретический справочник.

К пособию прилагается **диск, содержащий 4 теста, не представленных в книге, для компьютерного тестирования**. Пособие предназначено для выпускников общеобразовательных учреждений, а также для учителей, которые могут использовать его в процессе подготовки учащихся к ЕГЭ.

Книга является частью **учебно-методического комплекса «Информатика и ИКТ. Подготовка к ЕГЭ»**, включающего также книги «Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением (CD-диск)», «Информатика и ИКТ. Подготовка к ЕГЭ. Системы счисления», «Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий», «Информатика и ИКТ. Подготовка к ЕГЭ. Элементы алгебры логики», «Информатика и ИКТ. Подготовка к ЕГЭ. Сборник задач по программированию».

ББК 74.263.2

ISBN 978-5-9966-0619-1

© ООО «Легион», 2014

Оглавление

Введение	5
Глава I Краткий теоретический справочник	9
§ 1. Системы счисления	9
1.1. Позиционные системы счисления	9
1.2. Перевод чисел в десятичную систему счисления	10
1.3. Перевод чисел из десятичной системы счисления	11
1.4. Перевод чисел из двоичной системы счисления в восьмеричную, шестнадцатеричную и обратно	14
1.5. Перевод чисел из восьмеричной системы счисле- ния в шестнадцатеричную и обратно	16
1.6. Арифметические операции в позиционных системах счисления	16
§ 2. Кодирование информации	19
2.1. Основные понятия	19
2.2. Условие Фано	20
§ 3. Дискретное (цифровое) представление информации	21
3.1. Единицы измерения информации и их производные	21
3.2. Информационный объем	21
3.3. Представление числовой информации	24
3.4. Кодирование текстовой информации	27
3.5. Графическая информация. Основные понятия	28
3.6. Представление звуковой информации	29
§ 4. Построение алгебры высказываний	30
4.1. Простые и составные высказывания. Высказывательные переменные	30
4.2. Основные логические связки	31
4.3. Логические операции над высказываниями	31
4.4. Формулы и их логические возможности	34

4.5. Равносильные формулы	34
4.6. Тавтологии и противоречия. Таблицы истинности	35
4.7. Построение формул по заданным таблицам истинности ..	35
4.8. Свойства логических операций (законы логики)	37
§ 5. Алгоритмы	40
5.1. Способы задания алгоритма	40
5.2. Основные алгоритмические конструкции	43
§ 6. Файловые системы	46
6.1. Основные понятия	46
6.2. Маски имён файлов	46
§ 7. Компьютерные сети	47
7.1. Основные понятия	47
7.2. Адресация в сети	48
7.3. Маска сети	48
Глава II Учебно-тренировочные тесты	50
§ 1. Учебно-тренировочные тесты	51
Вариант № 1	51
Вариант № 2	66
Вариант № 3	82
Вариант № 4	97
Вариант № 5	114
Вариант № 6	128
Вариант № 7	141
Вариант № 8	157
Вариант № 9	172
Вариант № 10	188
Вариант № 11	203
Вариант № 12	217
§ 2. Ответы к тестам	232
Литература	270

Введение

С 2010 года во многих вузах в качестве одного из вступительных экзаменов введён предмет «Информатика и ИКТ». Ранее ЕГЭ по информатике проводился далеко не во всех регионах. Возросшим интересом к изучению этой дисциплины и необходимостью подготовки к экзамену и продиктован выпуск данного пособия. Книга, которую вы держите в руках, предназначена для учащихся выпускных классов средних общеобразовательных учреждений и их учителей.

Книга **«Информатика и ИКТ. Подготовка к ЕГЭ-2015. Пособие с электронным приложением (CD-диск)»** входит в **учебно методический комплекс «Информатика и ИКТ. Подготовка к ЕГЭ»**, состоящий из книг: «Информатика и ИКТ. Подготовка к ЕГЭ-2015», «Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением (CD-диск)», «Информатика и ИКТ. Подготовка к ЕГЭ. Системы счисления», «Информатика и ИКТ. Подготовка к ЕГЭ. Элементы алгебры логики», «Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий», «Информатика и ИКТ. 9–11 классы. Карманный справочник», «Информатика и ИКТ: основы программирования. 9–11 классы. Карманный справочник».

Краткое содержание этих книг приведено ниже.

Автором были проанализированы все задания демонстрационных вариантов ЕГЭ 2005–2014 гг. На основе ключевых идей этих тестов составлены 16 вариантов тестовых заданий (12 из которых представлены в книге и 4 варианта на диске). Каждый вариант экзаменационной работы состоит из двух частей и включает в себя 27 заданий, различающихся формой и уровнем сложности.

Часть 1 содержит 23 задания с кратким ответом. В этой части предложены следующие разновидности заданий с кратким ответом:

— задания на выбор и запись одного или нескольких правильных ответов из предложенного перечня ответов;

- задания на вычисление определённой величины;
- задания на установление правильной последовательности, представленной в виде строки символов по определённому алгоритму.

Часть 2 состоит из 4 заданий, для выполнения которых необходимо написать развёрнутый ответ в произвольной форме.

В книге приводятся ответы ко всем заданиям части 1 и решения всех заданий части 2.

Варианты, представленные на диске, содержат задания только части 1. После прохождения этих тестов можно сразу получить информацию о верно и неверно решённых задачах.

Методические рекомендации. Многочисленные опросы выпускников, сдающих ЕГЭ, позволяют сделать вывод о том, что основная подготовка к экзамену осуществляется на уроках информатики. Мы напомним истину, заключающуюся в том, что *необходимо добиваться от учащихся не формального усвоения программного материала, но его глубоко осознанного понимания*. Разумеется, в процессе преподавания необходимо делать определённые акценты на те разделы, которые представлены в тестах ЕГЭ.

Для подготовки к экзамену по информатике и ИКТ мы предлагаем следующую схему:

1. Для начала обратитесь к книге **«Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением (CD-диск)»**. Прорешайте тест, который дан в начале книги. При выполнении теста не задерживайтесь на заданиях, вызывающих затруднения, а отметьте их. Это поможет выявить темы, по которым недостаточно знаний для успешного решения задач.

2. На основе сделанных заметок перейдите к изучению тем, по которым знаний недостаточно для решения предлагаемых задач. Для отработки таких тем рекомендуем обратиться к тому же пособию **«Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением»**. В этой книге в начале каждой темы представлены типовые задания с решениями, далее подобные задачи для самостоятельного решения. Прорешав задачи, которые вызывают наибольшие затруднения, переходите к выполнению тематических тестов по соответствующей теме. Задания в этих тестах расположены по мере возрастания уровня сложности.

3. Для проверки степени усвоения тематического материала рекомендуем обратиться к тестам, **содержащимся на диске**, прилагаемом к кни-

ге **«Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением»**. После прохождения тестов можно сразу получить информацию о верно и неверно решённых задачах. И в случае затруднений при решении отдельных заданий следует снова обратиться к тематическим тестам. Для отработки навыков программирования советуем воспользоваться книгой **«Информатика и ИКТ. Сборник задач по программированию. Подготовка к ЕГЭ»**.

В случае возникновения затруднений при решении задач по темам: «Системы счисления», «Элементы алгебры логики», «Поиск выигрышных стратегий», можно обратиться к соответствующим книгам: **«Информатика и ИКТ. Подготовка к ЕГЭ. Системы счисления»**, **«Информатика и ИКТ. Подготовка к ЕГЭ. Элементы алгебры логики»**, **«Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий»**. Эти книги содержат только типовые тематические задачи с решениями.

4. После отработки всех тем переходите к выполнению тестов данной книги, а также тестов, представленных на диске.

5. Далее при необходимости можно снова обратиться к тематическим тестам, если возникнут проблемы при решении отдельных заданий.

6. Когда все темы изучены, следует выполнить следующие варианты тестовых заданий данной книги, фиксируя время и пытаясь уложиться в отведённый временной промежуток.

При самостоятельном решении задач вам помогут **карманные справочники «Информатика и ИКТ. 9–11 классы»** и **«Информатика и ИКТ: основы программирования. 9–11 классы»**. Обращение к справочному материалу даст возможность закрепить теоретический материал, необходимый для выполнения заданий.

Краткое содержание книг учебно-методического комплекса «Информатика и ИКТ. Подготовка к ЕГЭ».

1) Настоящая книга **«Информатика и ИКТ. Подготовка к ЕГЭ-2015. Пособие с электронным приложением (CD-диск)»** содержит 16 авторских учебно-тренировочных тестов, составленных по актуальной спецификации ЕГЭ и с учётом прошедшего экзамена, а также экзаменов прошлых лет. В книге приводятся ответы ко всем заданиям части 1 и решения всех заданий части 2. Варианты, представленные на диске, содержат задания только части 1. Данные здесь задания не включены в книгу. После прохождения этих тестов можно сразу получить информацию о верно и неверно решённых задачах.

2) Книга **«Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением (CD-диск)»** поможет при отработке необходимых школьнику тем. В этой книге в начале каждой темы представлены типовые задания с решениями, а далее подобные задачи для самостоятельного решения. Авторы старались включить в каждый тематический блок не только задачи, которые встречались ранее в ЕГЭ, но и вспомогательные задачи: более лёгкие для адаптации к материалу и сложные для закрепления изученного материала. Варианты, представленные на диске, содержат задания, соответствующие только части 1. После прохождения этих тестов можно сразу получить информацию о верно и неверно решённых задачах.

3) Книга **«Информатика и ИКТ. Сборник задач по программированию. Подготовка к ЕГЭ»** содержит краткое изложение базовых конструкций языка программирования Паскаль, разбор некоторых алгоритмов и задачи по программированию, разделённые на тематические блоки. Настоящая книга окажет помощь при изучении базовых конструкций программирования.

4) Книги **«Информатика и ИКТ. Подготовка к ЕГЭ. Системы счисления»**, **«Информатика и ИКТ. Подготовка к ЕГЭ. Элементы алгебры логики»**, **«Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий»** содержат только типовые тематические задачи с решениями (задачи для самостоятельного решения в них отсутствуют).

5) Карманные справочники **«Информатика и ИКТ. 9–11 классы»** и **«Информатика и ИКТ: основы программирования. 9–11 классы»** содержат необходимый справочный материал для самостоятельной подготовки к ЕГЭ по информатике.

Желаем успехов!

Замечания и пожелания, касающиеся данной книги, можно присылать по электронной почте: legionrus@legionrus.com.

Обсудить пособие, оставить замечания и предложения, задать вопросы можно на форумах издательства

<http://legion-posobiya.livejournal.com>; f.legionr.ru.

Краткий теоретический справочник

§ 1. Системы счисления

Определение. Система счисления — это способ наименования и представления чисел с помощью символов. Такие символы в любой системе счисления называются цифрами.

Определение. Алфавит системы счисления — это совокупность символов, используемых в данной системе счисления.

Все системы счисления подразделяются на два класса — позиционные и непозиционные.

В непозиционных системах счисления от положения цифры в записи числа не зависит величина, которую она обозначает.

1.1. Позиционные системы счисления

В позиционных системах счисления величина, обозначаемая цифрой в записи числа, зависит от её позиции. Количество различных цифр p , используемых в позиционной системе, определяет название системы счисления и называется основанием p -й системы счисления. Например, система счисления, в основном применяемая в современной математике, является позиционной десятичной системой, её основание равно десяти.

Любое число N в позиционной системе счисления с основанием p может быть представлено в виде многочлена от p :

$$N = a_k p^k + a_{k-1} p^{k-1} + \dots + a_1 p^1 + a_0 p^0 + a_{-1} p^{-1} + a_{-2} p^{-2} + \dots,$$

где N — число, p — основание системы счисления ($p > 1$), a_i — цифры числа (коэффициенты при степенях p).

Числа в p -й системе счисления записывают в виде последовательности цифр:

$$N = a_k a_{k-1} \dots a_1 a_0, a_{-1} a_{-2} \dots$$

Запятая в последовательности отделяет целую часть числа от дробной (коэффициенты при неотрицательных степенях от коэффициентов при отрицательных степенях).

1.1.1. Двоичная система счисления

В двоичной системе используются две цифры: 0 и 1. В этой системе любое число может быть представлено в виде

$N = a_k a_{k-1} \dots a_1 a_0, a_{-1} a_{-2} \dots$, где a_i принимает значения либо 0, либо 1. Эта запись соответствует сумме степеней числа 2, взятых с указанными коэффициентами:

$$N = a_k 2^k + a_{k-1} 2^{k-1} + \dots + a_1 2^1 + a_0 2^0 + a_{-1} 2^{-1} + a_{-2} 2^{-2} + \dots$$

Например,

$$1011101,01 = 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2}.$$

1.1.2. Восьмеричная система счисления

В восьмеричной системе используется восемь цифр — 0, 1, 2, 3, 4, 5, 6, 7.

Число N в восьмеричной системе счисления может быть представлено в виде

$$N = a_k 8^k + a_{k-1} 8^{k-1} + \dots + a_1 8^1 + a_0 8^0 + a_{-1} 8^{-1} + a_{-2} 8^{-2} + \dots$$

Например,

$$63401,1 = 6 \cdot 8^4 + 3 \cdot 8^3 + 4 \cdot 8^2 + 0 \cdot 8^1 + 1 \cdot 8^0 + 1 \cdot 8^{-1}.$$

1.1.3. Шестнадцатеричная система счисления

Для обозначения цифр в шестнадцатеричной системе используется десять цифр — 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 и латинские буквы — A (10), B (11), C (12), D (13), E (14) и F (15).

Число N в шестнадцатеричной системе счисления может быть представлено в виде

$$N = a_k 16^k + a_{k-1} 16^{k-1} + \dots + a_1 16^1 + a_0 16^0 + a_{-1} 16^{-1} + a_{-2} 16^{-2} + \dots$$

Например,

$$A0D4 = 10 \cdot 16^3 + 0 \cdot 16^2 + 13 \cdot 16^1 + 4 \cdot 16^0.$$

1.2. Перевод чисел в десятичную систему счисления

Для того чтобы перевести число в десятичную систему, необходимо составить сумму степенного ряда с основанием системы, в которой записано число, а затем найти значение этой суммы.

Пример 1.1. Переведите число 110110,01 из двоичной системы в десятичную.

$$\begin{aligned} \text{Решение. } 110110,01_2 &= 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + \\ &+ 1 \cdot 2^1 + 0 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = 32 + 16 + 4 + 2 + 0,25 = \\ &= 54,25_{10}. \end{aligned}$$

Ответ: $54,25_{10}$.

Пример 1.2. Переведите число 206,4 из восьмеричной системы в десятичную.

Решение. $206,4_8 = 2 \cdot 8^2 + 0 \cdot 8^1 + 6 \cdot 8^0 + 4 \cdot 8^{-1} = 128 + 6 + 0,5 = 134,5_{10}$.

Ответ: $134,5_{10}$.

Пример 1.3. Переведите число $A2F,4$ из шестнадцатеричной системы в десятичную.

Решение. $A2F,4_{16} = 10 \cdot 16^2 + 2 \cdot 16^1 + 15 \cdot 16^0 + 4 \cdot 16^{-1} = 2560 + 32 + 15 + 0,25 = 2607,25_{10}$.

Ответ: $2607,25_{10}$.

1.3. Перевод чисел из десятичной системы счисления

Способ 1.

Выполнить действия по следующему алгоритму:

1. Представить десятичное число в виде:

$$N = a \cdot p^k + m,$$

где p — основание системы счисления ($p > 1$), a ($a < p$) и k — наибольшие числа, при которых $a \cdot p^k < N$, m — остаток от деления N на $a \cdot p^k$.

2. Если $m > p$, выполнить действия п. 1 для числа m .

3. Если $k \neq 0$, в k -й позиции числа записать соответствующий коэффициент a . Если $k = 0$, то в k -й позиции числа записать 0. Если $m \neq 0$, записать m в нулевой позиции.

Способ 2.

Выполнить последовательное деление десятичного числа и затем получаемых целых частных на основание той системы, в которую оно переводится, до тех пор, пока не получится частное, меньшее делителя. Число в новой системе записывается в виде остатков от деления, начиная с последнего.

Пример 1.4. Переведите число 344 из десятичной системы в двоичную.

Решение.

Способ 1.

$$344 = 2^8 + 88; 88 = 2^6 + 24; 24 = 2^4 + 8; 8 = 2^3.$$

Позиция 8 7 6 5 4 3 2 1 0

Коэффициент 1 0 1 0 1 1 0 0 0

Ответ: 101011000_2 .

Способ 2.

$$\begin{array}{r}
 344 \overline{) 2} \\
 344 \overline{) 172} \overline{) 2} \\
 \underline{0} \quad 172 \overline{) 86} \overline{) 2} \\
 \underline{0} \quad 86 \overline{) 43} \overline{) 2} \\
 \underline{0} \quad 42 \overline{) 21} \overline{) 2} \\
 \underline{1} \quad 20 \overline{) 10} \overline{) 2} \\
 \underline{1} \quad 10 \overline{) 5} \overline{) 2} \\
 \underline{0} \quad 4 \overline{) 2} \overline{) 2} \\
 \underline{1} \quad 2 \overline{) 1} \\
 \underline{0}
 \end{array}$$

Ответ: 101011000_2 .

Пример 1.5. Переведите число 936 из десятичной системы в шестнадцатеричную.

Решение.

Способ 1.

$$936 = 3 \cdot 16^2 + 168; 168 = 10 \cdot 16^1 + 8.$$

Позиция 2 1 0

Коэффициент 3 A 8

Ответ: $3A8_{16}$.

Способ 2.

$$\begin{array}{r}
 936 \overline{) 16} \\
 928 \overline{) 58} \overline{) 16} \\
 \underline{8} \quad 48 \overline{) 3} \\
 \underline{10}
 \end{array}$$

Ответ: $3A8_{16}$.

1.3.1. Перевод правильной десятичной дроби из десятичной системы счисления

Для того чтобы перевести правильную десятичную дробь из десятичной системы счисления в другую, необходимо последовательно умножать эту дробь, а затем получаемые дробные части на основание той системы, в которую она переводится.

Умножение производится до тех пор, пока дробная часть не станет равной нулю или будет достигнута требуемая точность.

В новой системе дробь записывается в виде целых частей произведений, начиная с первого.

Пример 1.6. Переведите число 0,532 из десятичной системы в двоичную с точностью до тысячных.

Решение.

$$\begin{array}{r} 0, \times \begin{array}{l} 532 \\ \hline 2 \\ 2 \end{array} \\ 1, \times \begin{array}{l} 064 \\ \hline 2 \\ 2 \end{array} \\ 0, \times \begin{array}{l} 128 \\ \hline 2 \\ 2 \end{array} \\ 0, \times \begin{array}{l} 256 \end{array} \end{array}$$

Ответ: $0,100_2$.

Пример 1.7. Переведите число 0,974 из десятичной системы в шестнадцатеричную с точностью до тысячных.

Решение.

$$\begin{array}{r} 0, \times \begin{array}{l} 974 \\ \hline 16 \\ 16 \end{array} \\ 15, \times \begin{array}{l} 584 \\ \hline 16 \\ 16 \end{array} \\ 9, \times \begin{array}{l} 344 \\ \hline 16 \\ 16 \end{array} \\ 5, \times \begin{array}{l} 504 \end{array} \end{array}$$

Ответ: $0, F95_{16}$.

1.3.2. Перевод смешанного числа из десятичной системы счисления

Для того чтобы перевести число, содержащее и целую, и дробную части, из десятичной системы счисления в другую, необходимо сначала перевести его целую часть, затем отдельно — дробную часть. В ответе перед запятой следует записать целую часть, а после запятой — дробную часть.

Пример 1.8. Переведите число 344,532 из десятичной системы в двоичную с точностью до тысячных.

Решение. Переводим целую часть числа (см. пример 1.4). Получаем $344_{10} = 101011000_2$. Переводим с указанной точностью его дробную часть (см. пример 1.6). Получаем $0,532_{10} = 0,100_2$. Дописываем после целой части дробную: $344,532_{10} = 101011000,100_2$.

Ответ: $101011000,100_2$.

Пример 1.9. Переведите число 936,974 из десятичной системы в шестнадцатеричную с точностью до тысячных.

Решение. Переводим целую часть числа (см. пример 1.5). Получаем: $936_{10} = 3A8_{16}$.

Переводим с указанной точностью дробную часть (см. пример 1.7).
Получаем: $0,974_{10} = 0, F95_{16}$.

Дописываем после целой части дробную: $936,974_{10} = 3A8, F95_{16}$.

Ответ: $3A8, F95_{16}$.

1.4. Перевод чисел из двоичной системы счисления в восьмеричную, шестнадцатеричную и обратно

1.4.1. Перевод чисел из двоичной системы счисления в восьмеричную и обратно

Для того чтобы перевести число из двоичной системы в восьмеричную, следует, двигаясь от запятой влево и вправо, разбить двоичное число на группы по три разряда, дополняя при необходимости нулями крайние левую и правую группы. Затем триаду заменить соответствующей восьмеричной цифрой.

После перевода числа из двоичной системы счисления в восьмеричную количество цифр уменьшится в 3 раза.

Цифра	0	1	2	3	4	5	6	7
Триада	000	001	010	011	100	101	110	111

Пример 1.10. Переведите число $10011001111,0101$ из двоичной системы в восьмеричную.

Решение.

$$\underbrace{010}_2 \underbrace{011}_3 \underbrace{001}_1 \underbrace{111}_7, \underbrace{010}_2 \underbrace{100}_4 = 2317,24_8$$

Ответ: $2317,24_8$.

Для перевода числа из восьмеричной системы в двоичную достаточно заменить каждую цифру этого числа соответствующим трёхразрядным двоичным числом (триадой), при этом отбрасывают незначащие нули в старших и младших (после запятой) разрядах.

После перевода числа из восьмеричной системы в двоичную количество цифр увеличивается в 3 раза.

Пример 1.11. Переведите число $204,4$ из восьмеричной системы в двоичную.

Решение.

$$\underbrace{2}_{010} \underbrace{0}_{000} \underbrace{4}_{100}, \underbrace{4}_{100} = 10000100,1_2$$

Ответ: $10000100,1_2$.

1.4.2. Перевод чисел из двоичной системы счисления в шестнадцатеричную и обратно

Для того чтобы перевести число из двоичной системы в шестнадцатеричную, следует, двигаясь от запятой влево и вправо, разбить двоичное число на группы по четыре разряда, дополняя при необходимости нулями крайние левую и правую группы. Затем тетраду заменить соответствующей шестнадцатеричной цифрой.

После перевода числа из двоичной системы счисления в шестнадцатеричную количество цифр уменьшится в 4 раза.

Цифра	0	1	2	3	4	5	6	7
Тетрада	0000	0001	0010	0011	0100	0101	0110	0111
Цифра	8	9	A	B	C	D	E	F
Тетрада	1000	1001	1010	1011	1100	1101	1110	1111

Пример 1.12. Переведите число 1011111011,100011 из двоичной системы в шестнадцатеричную.

Решение.

$$\underbrace{0101}_5 \underbrace{1111}_F \underbrace{1011}_B, \underbrace{1000}_8 \underbrace{1100}_C = 5FB,8C_{16}$$

Ответ: 5FB,8C₁₆.

Для перевода числа из шестнадцатеричной системы в двоичную достаточно заменить каждую цифру этого числа соответствующим четырёхразрядным двоичным числом (тетрадой), при этом отбрасывают незначащие нули в старших и младших разрядах.

После перевода числа из шестнадцатеричной системы в двоичную количество цифр увеличится в 4 раза.

Пример 1.13. Переведите число 6C3,A из шестнадцатеричной системы в двоичную.

Решение.

$$\underbrace{6}_{0110} \underbrace{C}_{1100} \underbrace{3}_{0011}, \underbrace{A}_{1010} = 11011000011,101_2$$

Ответ: 11011000011,101₂.

1.5. Перевод чисел из восьмеричной системы счисления в шестнадцатеричную и обратно

Перевод из восьмеричной системы счисления в шестнадцатеричную и обратно осуществляется через двоичную систему с помощью триад и тетрад.

Пример 1.14. Переведите число 135,14 из восьмеричной системы счисления в шестнадцатеричную.

Решение.

$$\underbrace{1}_{001} \underbrace{3}_{011} \underbrace{5}_{101}, \underbrace{1}_{001} \underbrace{4}_{100} = 1011101,0011_2 = \underbrace{0101}_5 \underbrace{1101}_D, \underbrace{0011}_3 = 5D,3_{16}$$

Ответ: 5D,3₁₆.

Пример 1.15. Запишите число 5731,56₈ в шестнадцатеричной системе счисления.

$$\text{Решение. } 5731,56_8 = 101111011001, 101110_2 = 101111011001, 10111000_2 = BD9,B8_{16}.$$

Ответ: BD9,B8₁₆.

1.6. Арифметические операции в позиционных системах счисления

1. При сложении чисел в произвольной позиционной системе счисления с основанием p в каждом разряде производится сложение цифр слагаемых и цифры, переносимой из соседнего младшего разряда, если она имеется. При этом необходимо учитывать, что если при сложении чисел получилось число, большее или равное p , то представляем его в виде $pk+b$, где $k \in N, b \in N_0, 0 \leq b \leq p-1$ — остаток от деления полученного числа на основание системы счисления. Число b является количеством единиц в данном разряде, а число k — количеством единиц переноса в следующий разряд.

Пример 1.15. Выполните сложение двоичных чисел

$$X = 1011,1, Y = 1101,01 \text{ и } Z = 11101,11.$$

[illegible]

Пример 1.16. Выполните сложение шестнадцатеричных чисел $X = 5A,B, Y = 9F3,C1$ и $Z = A58,F$.

$$\begin{array}{r}
 1112 \\
 + 5A,B \\
 + 9F3,C1 \\
 \hline
 A58.F \\
 \hline
 14A7,61
 \end{array}$$

$$\begin{array}{r|l}
 \begin{array}{r}
 \text{1} \\
 \hline
 9+10+1=20=I \cdot 16+4 \\
 5+15+5+1=26=I \cdot 16+10
 \end{array}
 &
 \begin{array}{r}
 \text{1} \\
 \hline
 11+12+15=38=2 \cdot 16+6 \\
 10+3+8+2=23=I \cdot 16+7
 \end{array}
 \end{array}$$

2. При вычитании чисел в p -ой системе счисления цифры вычитаются поразрядно. Если в рассматриваемом разряде необходимо от меньшего числа отнять большее, то занимается единица следующего (большого) разряда. Занимаемая единица равна p единицам этого разряда (аналогично, когда мы занимаем единицу в десятичной системе счисления, то занимаемая единица равна 10).

Ответ: 10001101,1₂.

$$\begin{array}{r}
 \text{.....} \\
 \begin{array}{r}
 11001001,01 \\
 - \quad 111011,11 \\
 \hline
 10001101,10
 \end{array} \\
 \begin{array}{l}
 \boxed{1-0=1} \quad \boxed{1-1=0} \\
 \boxed{0-0=0} \quad \boxed{2-1=1} \\
 \boxed{2-1=0} \quad \boxed{2-1=1} \\
 \boxed{2-1=0} \quad \boxed{2-1=1} \\
 \boxed{2-1=1} \quad \boxed{2-1=1}
 \end{array}
 \end{array}$$

а)

$$\begin{array}{r}
 \text{..} \\
 \begin{array}{r}
 C9,4 \\
 - \quad 3B,C \\
 \hline
 8D,8
 \end{array} \\
 \begin{array}{l}
 \boxed{16+4-12=8} \\
 \boxed{16+(9-1)-11=13=D} \\
 \boxed{(12-1)-3=8}
 \end{array}
 \end{array}$$

б)

Рис. 1.

Пример 1.18. Найдите разность шестнадцатеричных чисел $C9,4 - 3B,C$.

Решение. См. рис. 1 б).

Ответ: $8D,8_{16}$.

3. При умножении чисел в p -ой системе счисления каждая цифра второго множителя умножается последовательно на цифру каждого из разрядов первого множителя (так же, как и в десятичной системе счисления). При этом необходимо учитывать, что если в результате умножения чисел получилось число, большее или равное p , то представляем его в виде $pk+b$, где $k \in N, b \in N_0, 0 \leq b \leq p-1$ (b — остаток от деления полученного числа на основание системы счисления p). Число b записываем в единицы данного разряда, а число k запоминаем и добавляем его к результату произведения в следующем разряде.

Полученные результаты умножения складываем согласно описанию, представленному в пункте 1, и отделяем количество знаков после запятой, равное сумме знаков после запятой у сомножителей.

Пример 1.19. Найдите произведение восьмеричных чисел $37,27 \cdot 4,6$.

Решение.

$$\begin{array}{r}
 \times 37,27 \\
 \quad \underline{4,6} \\
 + 27 \ 412 \\
 \hline
 175 \ 34 \\
 \hline
 224,752
 \end{array}
 \quad
 \begin{array}{l}
 7 \cdot 6 = 42 = 5 \cdot 8 + 2 \\
 2 \cdot 6 + 5 = 17 = 2 \cdot 8 + 1 \\
 7 \cdot 6 + 2 = 44 = 5 \cdot 8 + 4 \\
 3 \cdot 6 + 5 = 23 = 2 \cdot 8 + 7
 \end{array}
 \quad
 \begin{array}{l}
 7 \cdot 4 = 28 = 3 \cdot 8 + 4 \\
 2 \cdot 4 + 3 = 11 = 1 \cdot 8 + 3 \\
 7 \cdot 4 + 1 = 29 = 3 \cdot 8 + 5 \\
 3 \cdot 4 + 3 = 15 = 1 \cdot 8 + 7
 \end{array}$$

$$\begin{array}{l}
 \boxed{7+5=12=1 \cdot 8+4} \\
 \boxed{2+7+1=10=1 \cdot 8+2} \\
 \boxed{1+1=2}
 \end{array}$$

Ответ: $224,752_8$.

4. Деление чисел в p -ой системе счисления производится так же, как и десятичных чисел, при этом используются правила умножения, сложения и вычитания чисел в p -ой системе счисления (см. пп. 1–3).

Пример 1.20. Найдите частное от деления $B2B, 8 : 4, C$ в шестнадцатеричной системе счисления.

Решение.

$$\begin{array}{r}
 \begin{array}{r|l}
 B2B8 & 4C \\
 \hline
 98 & 25A \\
 \hline
 1AB & \\
 \hline
 17C & \\
 \hline
 2F8 & \\
 \hline
 2F8 & \\
 \hline
 0 &
 \end{array}
 \end{array}$$

Ответ: $25A_{16}$.

§ 2. Кодирование информации

2.1. Основные понятия

Кодом называют правило (или совокупность правил), в соответствии с которым производится отображение дискретных сообщений сигналами в виде определённых сочетаний символов вторичного алфавита.

Кодирование — это перевод информации, представленной символами первичного алфавита, в последовательность кодов.

Декодирование (операция, обратная кодированию) — перевод последовательности кодов в соответствующий набор символов первичного алфавита.

Операции кодирования и декодирования называются *обратимыми*, если их последовательное применение не приводит к потере информации.

Код является *однозначно декодируемым*, если любое слово, составленное из кодовых слов, можно декодировать только единственным способом.

По условию построения кодовых комбинаций коды делят на *равномерные* и *неравномерные*.

В равномерных кодах все сообщения передаются кодовыми группами с одинаковым числом элементов.

Равномерное кодирование всегда допускает однозначное декодирование.

Пример 2.1. Для кодирования букв *Г, Н, О, Ь* использовали четырёхразрядные двоичные числа, начинающиеся и оканчивающиеся на 1 (от 1001 до 1111 соответственно). В результате кодирования некоторого сообщения получили последовательность 11011001110110111111.

Определите закодированное сообщение.

Решение. Из условия следует, что заданные буквы имеют следующие коды:

Г — 1001, *Н* — 1011, *О* — 1101, *Ь* — 1111.

Для расшифровки достаточно разбить кодовую последовательность 11011001110110111111 на четвёрки: 1101 1001 1101 1011 1111 — и на месте каждой из них записать соответствующую букву.

Получим сообщение *ОГОНЬ*.

Ответ: ОГОНЬ.

Префиксный код — код, в котором никакое кодовое слово не является началом (префиксом) другого.

Аналогично можно определить **постфиксный код** — это код, в котором никакое кодовое слово не является концом (постфиксом) другого.

2.2. Условие Фано

Условие Фано: неравномерный код может быть однозначно декодирован, если никакой из кодов не совпадает с началом (префиксом) какого-либо другого, более длинного кода.

Декодирование для префиксных кодов однозначно и выполняется слева направо.

Пример 2.2.

1) Составить префиксный двоичный код для кодирования сообщения, состоящего из букв *а, е, л, н, п, т*.

2) Используя полученные коды, декодировать сообщение 100010000111110100.

Решение. 1) Поставим в соответствие буквам *а* и *е* коды 00 и 11. Для выполнения условия Фано коды оставшихся букв должны начинаться либо с 01, либо с 10. Например: *л* — 010, *н* — 011, *п* — 100, *т* — 101.

2) Для декодирования заданного сообщения будем отыскивать входящие в него последовательности полученных кодов и на месте каждой из найденных последовательностей запишем соответствующую букву. Так как для кодирования использовался префиксный код, то поиск будем осуществлять слева направо.

1 и 10 не являются кодами ни одной из букв. Первые три цифры — 100 — являются кодом буквы *п*. Три следующие — 010 — код буквы *л*

(букв с кодами 0 и 01 нет). Просматривая последовательность далее, получим разбиение:

100 010 00 011 11 101 00

п л а н е т а

Пример 2.3. Для кодирования используется неравномерный двоичный код: А – 01, Б – 101, В – 110, Г – 001, Д – 111. Определите, является ли этот код однозначно декодируемым. Можно ли сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было декодировать однозначно? Коды остальных букв меняться не должны.

Решение. Данный код является префиксным. Если код буквы Б поменять на 10, то коды букв А – 01, Б – 10, В – 110, Г – 001, Д – 111 по-прежнему будут образовывать префиксный код, и, следовательно, он однозначно декодируем.

§ 3. Дискретное (цифровое) представление информации

3.1. Единицы измерения информации и их производные

Бит — минимальная единица количества информации, равная одному двоичному разряду.

Байт — единица количества информации, являющаяся наименьшей единицей памяти компьютера и равная 8 битам.

Для больших объёмов информации используют производные единицы измерения:

1 б (байт) = 8 бит (8 двоичных разрядов)

1 Кб (килобайт) = 2^{10} б = 1024 б

1 Мб (мегабайт) = 2^{20} б = 1024 Кб

1 Гб (гигабайт) = 2^{30} б = 1024 Мб

1 Тб (терабайт) = 2^{40} б = 1024 Гб

1 Пб (петабайт) = 2^{50} б = 1024 Тб

3.2. Информационный объём

Существует несколько подходов к измерению информации. Выделим два из них.

3.2.1. Алфавитный (технический) подход

В технике информацией, как правило, считается любая последовательность знаков или символов. Для определения количества такой информации подсчитывают длину такой последовательности (сообщения) без учёта её содержательной части.

Определение. Информационным объёмом сообщения называется количество двоичных символов, которое используется для кодирования этого сообщения.

Формула Хартли. Пусть M — количество символов (мощность) алфавита, в котором записано сообщение, N — количество символов в записи сообщения. Тогда информационный объём сообщения

$$I = N \cdot \log_2 M \quad (1)$$

Если $\log_2 M$ не является целым числом, то его нужно округлить в большую сторону или найти значение $\log_2 \tilde{M}$, где $\tilde{M}$ — ближайшая целая степень 2, $\tilde{M} > M$.

Информационный объём сообщения, выраженный в битах, и минимальное количество разрядов, необходимое для записи сообщения в двоичном алфавите, совпадают.

Чем длиннее код, тем больше информации он содержит.

Одним битом могут быть выражены два различных значения: 0 или 1 (да или нет, чёрное или белое, истина или ложь и так далее). Двумя битами можно закодировать четыре различных значения: 00 01 10 11. Тремя битами можно закодировать восемь различных значений: 000 001 010 011 100 101 110 111.

С помощью n двоичных разрядов можно закодировать двоичным кодом все элементы множества мощностью 2^n . Информационный объём одного символа алфавита, обозначающего элемент данного множества, равен n .

Пример 3.1. Определите информационный объём слова «разряд», если считать, что алфавит состоит из 10 букв.

Решение. Длина данного сообщения равна 6, мощность алфавита равна 10. По формуле (1) находим $I = 6 \cdot \log_2 10$. Так как число 10 не является целой степенью числа 2, то значение $\log_2 10$ необходимо округлить в большую сторону или найти значение $\log_2 \tilde{M}$, где $\tilde{M}$ — ближайшая целая степень числа 2, $\tilde{M} > 10$. Следовательно, $\tilde{M} = 16$. Тогда $I = 6 \cdot \log_2 16 = 6 \cdot 4 = 24$ бита.

Ответ: 24.

Пример 3.2. Какое количество информации необходимо для кодирования каждого символа из 256 символов некоторого алфавита?

Решение. По формуле (1) находим $I = 1 \cdot \log_2 256 = 8$ битов.

Ответ: 8 битов.

3.2.2. Вероятностный подход

Определение. Количество информации можно рассматривать как меру уменьшения неопределённости знания при получении информационных сообщений.

За единицу количества информации принимается такое количество информации, которое содержится в информационном сообщении, уменьшающем неопределённость знания в два раза. Такая единица названа битом.

Пусть N — общее число возможных исходов какого-то процесса, и из них интересующее нас событие может произойти K раз. Тогда вероятность этого события равна K/N . Вероятность выражается в долях единицы.

Формула Шеннона. Количество информации для событий с различными вероятностями определяется по формуле

$$I = - \sum_{i=1}^N p_i \log_2 p_i, \quad (2)$$

где I — количество информации; N — количество возможных событий; p_i — вероятности отдельных событий.

Если события равновероятны, то количество информации определяется по формуле

$$I = \log_2 N \quad (3)$$

или из уравнения

$$N = 2^I. \quad (4)$$

Пример 3.3. В корзине лежат 8 мячей разного цвета (красный, синий, жёлтый, зелёный, оранжевый, фиолетовый, белый, коричневый). Какое количество информации несёт в себе сообщение о том, что из корзины будет вынут мяч красного цвета?

Решение. Так как возможности вынуть мяч каждого из имеющихся цветов равновероятны, то для определения количества информации, содержащегося в сообщении о выпадении мяча красного цвета, воспользуемся формулой (3): $I = \log_2 N = \log_2 8 = 3$ (бита).

Ответ: 3 бита.

Пример 3.4. В корзине лежат 16 мячей разного цвета: 4 красных, 8 синих, 4 жёлтых. Какое количество информации несёт в себе сообщение о том, что из корзины извлечён один мяч?

Решение. Так как количество мячей различных цветов неодинаково, то вероятности зрительных сообщений о цвете вынутого мяча различны. Для определения этих вероятностей разделим количество мячей одного цвета на общее количество мячей. Получим вероятность вынуть мяч красного цвета $p_k = 4/16 = 0,25$, синего цвета $p_c = 8/16 = 0,5$, жёлтого цвета $p_{ж} = 4/16 = 0,25$.

Так как события не являются равновероятными, то воспользуемся формулой (2):

$$\begin{aligned} I &= -(p_k \log_2 p_k + p_c \log_2 p_c + p_{ж} \log_2 p_{ж}) = \\ &= -(0,25 \log_2 0,25 + 0,5 \log_2 0,5 + 0,25 \log_2 0,25) = \\ &= -(2 \cdot 0,25 \cdot (-2) + 0,5 \cdot (-1)) = 1,5 \text{ (бита)}. \end{aligned}$$

Ответ: 1,5 бита.

Количество информации, содержащейся в алфавитном сообщении

Если алфавит состоит из N символов, то количество информации, которое несёт один символ, можно определить по формуле (2), или в случае, если считать, что появление каждого символа — события равновероятные, — по формулам (3–4).

Чтобы определить количество информации, содержащейся в сообщении, записанном в некотором алфавите, следует количество информации, которое несёт в себе один символ этого алфавита, умножить на число символов в сообщении.

Пример 3.5. Известно, что объём сообщения составляет 3 Кб. Определите мощность алфавита, с помощью которого записано это сообщение, если известно, что оно содержит 3072 символа.

Решение. Объём данного сообщения равен $3 \text{ Кб} = 3 \cdot 1024 \cdot 8 \text{ бит} = 24576 \text{ бит}$. Тогда на один символ приходится $24576 : 3072 = 8$. По формуле (4) определяем количество символов в рассматриваемом алфавите: $N = 2^I = 2^3 = 8$.

Ответ: 8 символов.

3.3. Представление числовой информации

Представление чисел в памяти компьютера имеет специфическую особенность, связанную с тем, что в памяти компьютера числа должны располагаться в байтах — минимальных по размеру адресуемых ячейках па-

мяти. Адресом числа считают адрес первого байта. В байте может содержаться произвольный код из восьми двоичных разрядов.

1. Целые числа представляются в памяти компьютера с фиксированной запятой. В этом случае каждому разряду ячейки памяти компьютера соответствует один и тот же разряд числа, запятая находится справа после младшего разряда (то есть вне разрядной сетки).

Для кодирования целых чисел от 0 до 255 достаточно иметь 8 разрядов двоичного кода (8 бит).

Десятичное число	Двоичный код
0	0000 0000
1	0000 0001
2	0000 0010
...	...
254	1111 1110
255	1111 1111

Для кодирования целых чисел от 0 до 65 535 требуется шестнадцать бит; 24 бита позволяют закодировать более 16,5 миллионов разных значений.

Если для представления целого числа в памяти компьютера отведено N бит, то количество различных значений будет равно 2^N .

Максимальное значение целого неотрицательного числа достигается в случае, когда во всех ячейках стоят единицы. Если под представление целого положительного числа отведено N бит, то максимальное значение будет равно $2^N - 1$.

Прямой код целого числа может быть получен следующим образом: число переводится в двоичную систему счисления, а затем его двоичную запись слева дополняют необходимым количеством незначащих нулей, соответствующим количеству незаполненных разрядов, отведённых для хранения числа.

2. Для представления целых чисел со знаком старший (левый) разряд отводится под знак числа. Если число положительное, то в знаковый разряд записывается 0, если число отрицательное, то 1.

Максимальное значение целого числа со знаком достигается в случае, когда в старшем разряде стоит 0, а во всех остальных ячейках стоят единицы. Если под представление целого числа со знаком отведено N бит, то максимальное значение будет равно $2^{N-1} - 1$. Поскольку количество возможных значений в N битах равно $2^N - 1$, то в случае представления целых чисел со знаком количество отрицательных значений на едини-

цу больше количества положительных значений. Такая ситуация связана с тем, что для представления нуля во всех ячейках стоят нули. Если же в знаковом разряде стоит единица, а во всех остальных разрядах — нули, то это представление соответствует отрицательному (как правило, наименьшему) числу.

Пример 3.6. Запишем вид числа -58 в памяти компьютера в 8-разрядном представлении.

Так как $58_{10} = 111010_2$, то число в памяти компьютера будет представлено следующим образом:

1	0	1	1	1	0	1	0
---	---	---	---	---	---	---	---

Представление в памяти компьютера целых положительных чисел совпадает с прямым кодом.

3. Другой способ представления целых чисел — **дополнительный код**.

Дополнительный код целого отрицательного числа может быть получен по следующему алгоритму:

- 1) записываем прямой код модуля числа;
- 2) инвертируем его (заменяем единицы нулями, нули единицами);
- 3) прибавляем к инверсному коду единицу.

Пример 3.7. Запишем дополнительный код числа -58 в 8-разрядном представлении.

1) Прямой код числа 58 есть 00111010; 2) инверсный (обратный) код 11000101; 3) дополнительный код 11000110.

4. При получении числа по его дополнительному коду необходимо определить его знак. Если число окажется положительным, то переводим его код в десятичную систему счисления.

В случае отрицательного числа необходимо выполнить следующий алгоритм:

- 1) вычитаем из кода числа 1;
- 2) инвертируем код;
- 3) переводим в десятичную систему счисления;
- 4) полученное число записываем со знаком минус.

Пример 3.8. Запишем число, соответствующее дополнительному коду 00110110.

Так как в старшем разряде данного числа ноль, то результат будет положительным. После перевода числа из двоичной системы счисления в десятичную получаем 54.

Пример 3.9. Запишем число, соответствующее дополнительному коду 10110110.

Так как в старшем разряде данного числа единица, то результат будет отрицательным. Вычитаем из кода единицу: $10110110 - 1 = 10110101$. Инвертируем код: 01001010. Переводим в десятичную систему счисления $01001010_2 = 74_{10}$. Полученное число записываем со знаком минус: -74_{10} .

3.4. Кодирование текстовой информации

1. Если с каждым символом алфавита сопоставить определённое целое число (например, порядковый номер), то с помощью двоичного кода можно кодировать текстовую информацию. Восьми двоичных разрядов достаточно для кодирования 256 различных символов. Этого вполне хватает, чтобы выразить все символы английского и русского языков, как строчные, так и прописные, а также знаки препинания, символы основных арифметических действий и некоторые общепринятые специальные символы.

2. Институт стандартизации США (ANSI — American National Standard Institute) ввёл в действие систему кодирования ASCII (American Standard Code for Information Interchange — стандартный код информационного обмена США). В системе ASCII закреплены таблицы кодирования некоторых специальных символов, строчных и прописных русских и латинских букв, цифр, знаков препинания, арифметических действий и других.

Каждому символу ASCII соответствует 8-битовый двоичный код, например, $A — 01000001$, $B — 01000010$, $C — 01000011$, $D — 01000100$ и т.д. Таким образом, при записи текстового файла на диск каждый символ текста хранится в памяти компьютера в виде набора из восьми нулей и единиц.

3. Windows-1251 — кодировка символов русского языка; используется на некоторых локальных компьютерах, работающих на платформе Windows. Каждому символу в этой кодировке соответствует 8-битовый двоичный код.

4. КОИ-8 (код обмена информацией, восьмизначный); встречается в компьютерных сетях на территории России и в российском секторе Интернета.

5. UNICODE — система, основанная на 16-разрядном кодировании символов. Шестнадцать разрядов позволяют обеспечить уникальные коды для 65536 различных символов. Каждому символу в этой кодировке соответствует 16-битовый (2-байтовый) двоичный код.

Пример 3.10. Определите количество информационного объёма выражения «Жёсткий диск», записанного в кодировке UNICODE.

Решение. В системе UNICODE каждый символ кодируется двумя байтами. В приведённом выражении 12 символов. Следовательно, информационный объём этого выражения равен $12 \cdot 2 = 24$ (байта).

Ответ: 24 байта.

Пример 3.11. Каждая страница текста состоит из 32 строк, в каждой строке по 64 символа. Определите максимальное количество страниц такого текста (без учёта символов форматирования), записанного в кодировке КОИ-8 на USB Flash drive (UFD) ёмкостью 512 Мб.

Решение. В кодировке КОИ-8 каждый символ кодируется одним байтом. Информационный объём страницы текста составляет $32 \cdot 64 = 2048$ байт. Ёмкость UFD 512 Мб = $512 \cdot 1024 \cdot 1024 = 536870912$ байт. Следовательно, максимальное количество страниц указанного текста (без учёта символов форматирования), записанного в кодировке КОИ-8 на UFD ёмкостью 512 Мб, равно $536870912 : 2048 = 262144$.

Ответ: 262144 страницы.

3.5. Графическая информация. Основные понятия

Пиксель — это минимальный участок изображения, для которого независимым образом можно задать цвет.

Глубина цвета — количество информации, которое используется для кодирования цвета точки изображения. Наиболее распространёнными глубинами цвета являются 4, 8, 16 и 24 бита на точку.

Количество различных цветов N и количество информации I , необходимое для кодирования каждой точки, связаны формулой $N = 2^I$.

Глубина цвета, I (бит)	Количество цветов в палитре, N
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
16 (High Color)	65536
24 (True color)	16777216
32 (True color)	4294967296

Разрешающая способность экрана — количество пикселей на единицу длины, dpi (dots per inch — «точка на дюйм»).

Расчёт объёма видеопамати.

Информационный объём требуемой для хранения изображения видеопамати можно рассчитать по формуле:

$$I_{\Pi} = I \cdot X \cdot Y,$$

где I_{Π} — информационный объём видеопамати в битах;

$X \cdot Y$ — количество точек изображения (X — количество точек по горизонтали, Y — по вертикали);

I — глубина цвета в битах на точку.

Пример 3.12. Определите необходимый объём видеопамати для графического режима с пространственным разрешением 1024×768 точек и глубиной цвета 24 бита.

Решение. $I_{\Pi} = I \cdot X \cdot Y = 24 \text{ бита} \cdot 1024 \cdot 768 =$
 $= 18\,874\,368 \text{ бит} = 2\,359\,296 \text{ байт} = 2\,304 \text{ Кбайт} =$
 $= 2,25 \text{ Мбайт}.$

Ответ: 2,25 Мбайт.

3.6. Представление звуковой информации

Звук представляет собой распространяющуюся в воздухе, воде или другой среде волну с непрерывно меняющейся интенсивностью и частотой.

Для того чтобы компьютер мог обрабатывать звук, непрерывный звуковой сигнал должен быть преобразован в цифровую дискретную форму с помощью *временной дискретизации*.

Частота дискретизации звука — это количество измерений уровня звукового сигнала за одну секунду.

Частота дискретизации звука может лежать в диапазоне от 8000 до 48 000 измерений уровня звукового сигнала за одну секунду.

Уровни звукового сигнала можно рассматривать как набор возможных состояний N , для кодирования которых необходимо определённое количество информации I , которое называется глубиной кодирования звука.

Глубина кодирования звука — это количество информации (количество бит), которое необходимо для кодирования дискретных уровней громкости цифрового звука.

Уровень цифрового звукового сигнала можно рассчитать по формуле:
 $N = 2^I.$

Пример 3.13. Пусть глубина кодирования звука составляет 16 битов, тогда количество уровней громкости звука: $N = 2^I = 2^{16} = 65536$.

Размер (в байтах) цифрового моноаудиофайла можно оценить по формуле: $A = D \cdot T \cdot I/8$,

где D — частота дискретизации звука за одну секунду, T — время звучания или записи звука (с), I — глубина кодирования звука (бит).

Пример 3.14. Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 64 кГц и 16-битной глубиной кодирования. Запись длится 3 минуты, её результаты записываются в файл, сжатие данных не производится. Определить размер полученного файла.

Решение.

Согласно условию задачи $D = 64000$ (изм. в с), $T = 3 \text{ мин} = 180 \text{ с}$, $I = 16$ бит. Следовательно,

$A = D \cdot T \cdot I/8 = 64000 \cdot 180 \cdot 16/8 = 23040000 \text{ б} \approx \approx 23 \text{ Мб}$. При четырёхканальной (квадро) записи объём памяти, необходимый для хранения данных одного канала, умножается на 4; $23 \text{ Мб} \cdot 4 = 92 \text{ Мб}$.

Ответ: 92 Мб.

§ 4. Построение алгебры высказываний

4.1. Простые и составные высказывания. Высказывательные переменные

Высказывание — это предложение, о котором имеет смысл утверждать, истинно оно или ложно. Таким образом, отличительной особенностью высказываний является возможность принимать одно из двух значений: истина — 1 или ложь — 0.

Высказывания могут быть *простыми* или *составными*.

Если в высказывании A нельзя выделить некоторую часть, которая сама является высказыванием и не совпадает по смыслу с высказыванием A , то A называется простым высказыванием. В противном случае высказывание A называется составным.

Простые высказывания (а в некоторых случаях и составные) будем обозначать прописными буквами латинского алфавита, а факт истинности или ложности высказывания $A = 1$ или $A = 0$. Буквы, обозначающие переменные высказывания, будем называть *высказывательными переменными*.

4.2. Основные логические связи

Конструирование составных высказываний из простых осуществляется при помощи связок (см. табл. I.1).

Таблица I.1. Основные логические связи

Связки	Обозначения	Название соответствующих операций
нет; не; неверно; ...	$\neg$ ($\bar{}$)	отрицание
и; а; но; ...	$\&$ ($\wedge$)	конъюнкция
или; либо; ...	$\vee$	дизъюнкция
следует; влечёт; если ..., то ...; тогда; вытекает ...	$\rightarrow$	импликация
эквивалентно; равносильно; если и только если; тогда и только тогда; в том и только в том случае; ...	$\sim$ ($\leftrightarrow$)	эквиваленция

4.3. Логические операции над высказываниями

Логическая операция — это способ построения сложного высказывания из данных высказываний, при котором значение истинности сложного высказывания полностью определяется значениями истинности исходных высказываний.

Во избежание неодинаковой трактовки смысла каждой из связок определим этот смысл следующими таблицами.

1. *Логическое отрицание* (инверсия) образуется из высказывания с помощью добавления частицы «не» к сказуемому или использования оборота речи «неверно, что...» (см. табл. I.2).

Обозначения логического отрицания: НЕ A , $\neg A$, $\bar{A}$, NOT A , A' .

Таблица I.2. Логическая связка $\neg$

A	$\neg A$
1	0
0	1

Из таблицы следует, что отрицание высказывания истинно, когда высказывание ложно, и ложно, когда высказывание истинно.

2. *Логическое умножение* (конъюнкция) образуется соединением двух высказываний в одно с помощью союза «и» (см. табл. I.3).

Обозначения логического умножения: A и B , $A \wedge B$, $A \& B$, $A \cdot B$, $A \text{ AND } B$.

Таблица I.3. Логическая связка $\&$

A	B	$A \& B$
1	1	1
1	0	0
0	1	0
0	0	0

Из таблицы следует, что конъюнкция двух высказываний истинна тогда и только тогда, когда оба высказывания истинны, и ложна тогда и только тогда, когда ложно хотя бы одно из высказываний.

Пример 4.1. Найдите значение логического выражения

$x_1 \wedge x_2 \wedge x_3 \wedge x_4 \wedge x_5$ для следующих значений логических переменных: $x_1 = 1, x_2 = 1, x_3 = 0, x_4 = 1, x_5 = 1$.

Решение. Подставим значения логических переменных в заданное выражение: $1 \wedge 1 \wedge 0 \wedge 1 \wedge 1 \cong 0$.

Ответ: 0.

3. *Логическое сложение* (дизъюнкция) образуется соединением двух высказываний в одно с помощью союза «или» (см. табл. I.4).

Обозначения логического сложения: A или B , $A \vee B$, $A \mid B$, $A + B$, $A \text{ OR } B$.

Таблица I.4. Логическая связка $\vee$

A	B	$A \vee B$
1	1	1
1	0	1
0	1	1
0	0	0

Из таблицы следует, что дизъюнкция двух высказываний истинна тогда и только тогда, когда хотя бы одно из высказываний истинно, и ложна тогда и только тогда, когда ложны оба высказывания.

Пример 4.2. Найдите значение логического выражения

$\neg x_1 \vee x_2 \vee x_3 \vee \neg x_4 \vee x_5$ для следующих значений логических переменных:
 $x_1 = 1, x_2 = 0, x_3 = 0, x_4 = 0, x_5 = 0$.

Решение. Подставим значения логических переменных в заданное выражение:

$$\neg 1 \vee 0 \vee 0 \vee \neg 0 \vee 0 \cong 0 \vee 0 \vee 0 \vee 1 \vee 0 \cong 1.$$

Ответ: 1.

4. *Логическое следование* (импликация) образуется соединением двух высказываний в одно с помощью оборота речи «если ..., то ...» (см. табл. I.5).

Обозначения логического следования: $A \rightarrow B, A \Rightarrow B$. Говорят: если A , то B ; A влечёт B ; B следует из A .

Таблица I.5. Логическая связка $\rightarrow$

A	B	$A \rightarrow B$
1	1	1
1	0	0
0	1	1
0	0	1

Из таблицы следует, что импликация двух высказываний ложна тогда и только тогда, когда из истинного высказывания следует ложное (когда истинная посылка влечёт ложное заключение).

5. *Логическое равенство* (эквиваленция) образуется соединением двух высказываний с помощью оборота речи «тогда и только тогда, когда...» (см. табл. I.6).

Обозначения логического равенства: $A \sim B, A \Leftrightarrow B, A = B$. Говорят: A тогда и только тогда, когда B .

Таблица I.6. Логическая связка $\sim$

A	B	$A \sim B$
1	1	1
1	0	0
0	1	0
0	0	1

4.4. Формулы и их логические возможности

Формулами называются

1) прописные буквы латинского алфавита, снабжённые, быть может, штрихами или индексами и обозначающие высказывания или высказывательные переменные;

2) если f и g — формулы, то выражения

$$\neg f, (f \wedge g), (f \vee g), (f \rightarrow g), (f \sim g)$$

также являются формулами.

Других формул, кроме тех, которые определены пунктами 1) и 2), нет.

Формулы будем обозначать буквами $f, g, q, F, G, Q \dots$

Если $A_1, A_2, \dots, A_n$ — все буквы, участвующие в записи формулы f , то будем писать: $f = f(A_1, A_2, \dots, A_n)$.

Например, $f(A) = \neg A$, $g(A_1, A_2, A_3) = (A_3 \rightarrow (A_2 \rightarrow A_1))$, $q(A, B, C) = ((A \vee B) \rightarrow C)$ и т. д. Для уменьшения количества скобок в формулах условимся считать, что связка $\neg$ сильнее, чем все остальные связки, $\wedge$ и $\vee$ — сильнее, чем $\rightarrow$ и $\sim$. Кроме того, внешние скобки будем иногда опускать.

Логической возможностью формулы $f(A_1, \dots, A_n)$ от высказывательных переменных $A_1, \dots, A_n$ называется всякий набор конкретных значений истинности для букв $A_1, \dots, A_n$.

Так, например, всякая формула от одной буквы имеет две логические возможности: 0 и 1. Всякая формула от двух букв имеет четыре логические возможности: (1, 1), (1, 0), (0, 1), (0, 0).

Таблица вида

1	1
1	0
0	1
0	0

называется таблицей логических возможностей для всякой формулы от 2 букв (высказывательных переменных) A и B .

4.5. Равносильные формулы

Определение. Пусть f и g — две формулы, а $A_1, \dots, A_n$ — все высказывательные переменные, входящие в запись хотя бы одной из этих формул. *Общей логической возможностью* формул f и g называется всякий набор конкретных значений истинности для высказывательных переменных $A_1, \dots, A_n$.

Можно определить понятие общей логической возможности для любого конечного числа формул.

Определение. Две формулы f и g называются *равносильными* ($f \equiv g$), если во всякой общей для f и g логической возможности f и g принимают одинаковые значения.

4.6. Тавтологии и противоречия. Таблицы истинности

Определение. Формула f называется *тождественно истинной* (*тождественно ложной*), или *тавтологией* (*противоречием*), и обозначается $f \equiv 1$ ($f \equiv 0$), если во всех логических возможностях она принимает одно и то же значение, равное 1 (равное 0). Запись $\models f$ означает, что f — тавтология.

Для любых двух формул f и g истинно утверждение

$$f \equiv g \iff \models (f \sim g).$$

Определение. Таблица, в которой приведён перечень всех логических возможностей формулы f (общих логических возможностей формул $f_1, \dots, f_n$) вместе с указанием значений f (значений $f_1, \dots, f_n$) в каждой логической возможности (общей логической возможности), называется таблицей истинности формулы f (формул $f_1, \dots, f_n$).

Пример 4.3. Составьте таблицу истинности для формулы $f(X, Y, Z) = \neg((Y \rightarrow Z \wedge X) \rightarrow Z)$.

Решение.

X	Y	Z	$Z \wedge X$	$Y \rightarrow Z \wedge X$	$(Y \rightarrow Z \wedge X) \rightarrow Z$	$\neg((Y \rightarrow Z \wedge X) \rightarrow Z)$
0	0	0	0	1	0	1
0	0	1	0	1	1	0
0	1	0	0	0	1	0
0	1	1	0	0	1	0
1	0	0	0	1	0	1
1	0	1	1	1	1	0
1	1	0	0	0	1	0
1	1	1	1	1	1	0

4.7. Построение формул по заданным таблицам истинности

Рассмотрим вначале решение этой задачи на примере. Пусть формула $F = F(A_1, A_2, A_3)$ от трёх высказывательных переменных задана таблицей истинности (см. табл. I.7).

Понятно, что существует бесконечно много равносильных формул алгебры высказываний, имеющих эту таблицу истинности. Укажем способ нахождения двух таких формул.

Таблица 1.7. Таблица истинности формулы от трёх высказывательных переменных

A_1	A_2	A_3	$F(A_1, A_2, A_3)$
1	1	1	1
1	1	0	0
1	0	1	1
1	0	0	0
0	1	1	0
0	1	0	0
0	0	1	1
0	0	0	0

Помечаем те строки таблицы, в которых $F(A_1, A_2, A_3)$ принимает значение, равное 1. Это строки 1, 3, 7. Для каждой строки (логической возможности) составим формулу, истинную только в этой логической возможности и ложную во всех остальных логических возможностях.

1-я строка — $A_1 \wedge A_2 \wedge A_3$

3-я строка — $A_1 \wedge \neg A_2 \wedge A_3$

7-я строка — $\neg A_1 \wedge \neg A_2 \wedge A_3$.

Если возьмём теперь дизъюнкцию всех этих формул, то это и будет искомой формулой

$$F = (A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge \neg A_2 \wedge A_3) \vee (\neg A_1 \wedge \neg A_2 \wedge A_3). \quad (1)$$

Рассмотрим другое решение этой задачи. Помечаем те строки таблицы, в которых $F(A_1, A_2, A_3)$ принимает значение, равное 0. Это строки 2, 4, 5, 6, 8. Для каждой логической возможности составим формулу, ложную только в этой логической возможности и истинную во всех остальных логических возможностях.

2-я строка — $\neg A_1 \vee \neg A_2 \vee A_3$

4-я строка — $\neg A_1 \vee A_2 \vee A_3$

5-я строка — $A_1 \vee \neg A_2 \vee \neg A_3$

6-я строка — $A_1 \vee \neg A_2 \vee A_3$

8-я строка — $A_1 \vee A_2 \vee A_3$.

Если теперь возьмём конъюнкцию этих формул, то это также будет искомой, то есть имеющей заданную таблицу истинности, формулой

$$F = (\neg A_1 \vee \neg A_2 \vee A_3) \wedge (\neg A_1 \vee A_2 \vee A_3) \wedge (A_1 \vee \neg A_2 \vee \neg A_3) \wedge (A_1 \vee \neg A_2 \vee A_3) \wedge (A_1 \vee A_2 \vee A_3). \quad (2)$$

Формулы (1) и (2) равносильны, так как имеют одну и ту же таблицу истинности. В данном случае удобнее строить формулу (1).

4.8. Свойства логических операций (законы логики)

Для любых логических формул F , G , Q истинны следующие равносильности.

1. Закон двойного отрицания:

$$\neg \neg F \equiv F.$$

2. Идемпотентность операций $\wedge$ и $\vee$:

$$F \wedge F \equiv F; F \vee F \equiv F.$$

3. Коммутативность операций $\wedge$ и $\vee$:

$$F \wedge G \equiv G \wedge F; F \vee G \equiv G \vee F.$$

4. Ассоциативность операций $\wedge$ и $\vee$:

$$F \wedge (G \wedge Q) \equiv (F \wedge G) \wedge Q; F \vee (G \vee Q) \equiv (F \vee G) \vee Q.$$

5. Дистрибутивные законы каждой из операций $\wedge$ и $\vee$ относительно другой:

$$F \wedge (G \vee Q) \equiv (F \wedge G) \vee (F \wedge Q); F \vee (G \wedge Q) \equiv (F \vee G) \wedge (F \vee Q).$$

6. Законы поглощения:

$$F \wedge (F \vee G) \equiv F; F \vee (F \wedge G) \equiv F.$$

7. Законы де Моргана:

$$\neg(F \wedge G) \equiv \neg F \vee \neg G; \neg(F \vee G) \equiv \neg F \wedge \neg G.$$

8. Закон исключённого третьего:

$$F \vee \neg F \equiv 1.$$

9. Закон противоречия:

$$F \wedge \neg F \equiv 0.$$

10. Свойства тавтологии и противоречия:

$$F \wedge 1 \equiv F; \quad F \vee 0 \equiv F,$$

$$F \vee 1 \equiv 1; \quad F \wedge 0 \equiv 0,$$

$$\neg 1 \equiv 0; \quad \neg 0 \equiv 1.$$

11. Закон контрапозиции:

$$F \rightarrow G \equiv \neg G \rightarrow \neg F.$$

12. Правило исключения импликации:

$$F \rightarrow G \equiv \neg F \vee G.$$

13. Правило исключения эквиваленции:

$$F \sim G \equiv (F \rightarrow G) \wedge (G \rightarrow F).$$

В примерах 4.4–4.6 числа, расположенные над знаками равносильности, соответствуют номерам перечисленных выше законов, которые были использованы при преобразовании формулы.

Пример 4.4. Приведите следующую формулу к более простому виду:
 $\neg(\neg X \wedge \neg Y) \vee ((X \rightarrow Y) \wedge X).$

$$\begin{aligned} & \text{Решение. } \neg(\neg X \wedge \neg Y) \vee ((X \rightarrow Y) \wedge X) \stackrel{(7)}{\cong} \\ & \cong \neg\neg X \vee \neg\neg Y \vee ((\neg X \vee Y) \wedge X) \stackrel{(1)}{\cong} \\ & \cong X \vee Y \vee (\neg X \wedge X) \vee (Y \wedge X) \stackrel{(9)}{\cong} X \vee Y \vee 0 \vee (Y \wedge X) \stackrel{(10)}{\cong} \\ & \cong X \vee Y \vee (Y \wedge X) \stackrel{(6)}{\cong} X \vee Y. \end{aligned}$$

Пример 4.5. С помощью равносильных преобразований доказать, что формула $\neg(\neg X \vee Y) \rightarrow ((X \vee Y) \rightarrow X)$ является тождественно истинной.

$$\begin{aligned} & \text{Решение. } \neg(\neg X \vee Y) \rightarrow ((X \vee Y) \rightarrow X) \stackrel{(7,1,12)}{\cong} \\ & \cong (X \wedge \neg Y) \rightarrow (\neg(X \vee Y) \vee X) \stackrel{(7)}{\cong} \\ & \cong (X \wedge \neg Y) \rightarrow ((\neg X \wedge \neg Y) \vee X) \stackrel{(5)}{\cong} \\ & \cong (X \wedge \neg Y) \rightarrow ((\neg X \vee X) \wedge (\neg Y \vee X)) \stackrel{(8)}{\cong} \\ & \cong (X \wedge \neg Y) \rightarrow (1 \wedge (\neg Y \vee X)) \stackrel{(12,10)}{\cong} \\ & \cong \neg(X \wedge \neg Y) \vee (\neg Y \vee X) \stackrel{(7,1)}{\cong} \neg X \vee Y \vee \neg Y \vee X \stackrel{(8)}{\cong} 1. \end{aligned}$$

Пример 4.6. Сколько различных наборов логических переменных x_1 , x_2 и x_3 удовлетворяют условию:

$$\neg(x_1 \equiv x_2) \wedge ((x_1 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3)) = 0?$$

Решение.

$$\begin{aligned} & \neg(x_1 \equiv x_2) \wedge ((x_1 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3)) \stackrel{(1)}{\cong} \\ & \cong \neg(x_1 \equiv x_2) \wedge \neg\neg((x_1 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3)) \stackrel{(7)}{\cong} \\ & \cong \neg(x_1 \equiv x_2) \wedge \neg(\neg(x_1 \wedge \neg x_3) \wedge \neg(\neg x_1 \wedge x_3)) \stackrel{(7,1)}{\cong} \\ & \cong \neg(x_1 \equiv x_2) \wedge \neg((\neg x_1 \vee x_3) \wedge (x_1 \vee \neg x_3)) \stackrel{(3)}{\cong} \\ & \cong \neg(x_1 \equiv x_2) \wedge \neg((\neg x_1 \vee x_3) \wedge (\neg x_3 \vee x_1)) \stackrel{(12)}{\cong} \\ & \cong \neg(x_1 \equiv x_2) \wedge \neg((x_1 \rightarrow x_3) \wedge (x_3 \rightarrow x_1)) \stackrel{(13)}{\cong} \\ & \cong \neg(x_1 \equiv x_2) \wedge \neg(x_1 \equiv x_3) \stackrel{(7)}{\cong} \\ & \cong \neg((x_1 \equiv x_2) \vee (x_1 \equiv x_3)). \end{aligned}$$

Таким образом, выражение принимает вид:

$$\neg((x_1 \equiv x_2) \vee (x_1 \equiv x_3)) = 0.$$

Применяя отрицание к обеим частям, получаем:

$$\neg\neg((x_1 \equiv x_2) \vee (x_1 \equiv x_3)) = \neg 0.$$

$$\text{Отсюда, } (x_1 \equiv x_2) \vee (x_1 \equiv x_3) = 1.$$

Это уравнение представляет собой дизъюнкцию двух высказываний. Дизъюнкция истинна только в том случае, когда хотя бы одно из высказываний, связанных дизъюнкцией, истинно. Значит, должно выполняться $(x_1 \equiv x_2) = 1$ или $(x_1 \equiv x_3) = 1$. Если $x_1 = x_2$, то высказывание $x_1 \equiv x_2$ истинно. Следовательно, при любых значениях x_3 истинно высказывание $(x_1 \equiv x_2) \vee (x_1 \equiv x_3)$.

Если $x_1 \neq x_2$, то высказывание $x_1 \equiv x_2$ ложно. Следовательно, высказывание $(x_1 \equiv x_2) \vee (x_1 \equiv x_3)$ истинно только в случае, когда $x_1 = x_3$. Таким образом, количество наборов логических переменных, удовлетворяющих заданному уравнению, равно: $2 \cdot 3 = 6$.

Ответ: 6.

§ 5. Алгоритмы

Алгоритм — заранее заданное возможному исполнителю точное предписание совершить определённую последовательность действий для получения решения задачи за конечное число шагов.

5.1. Способы задания алгоритма

На практике наиболее распространены следующие способы задания алгоритмов:

- *словесный* (запись на естественном языке);
- *графический* (изображения из графических символов);
- *псевдокод* (полуформализованное описание алгоритмов на условном алгоритмическом языке, включающее в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические обозначения и др.);
- *программный* (тексты на языках программирования).

Словесный способ

Словесный способ записи алгоритмов представляет собой описание последовательных этапов обработки данных. Алгоритм задаётся в произвольном изложении на естественном языке.

Пример 5.1. Запишите алгоритм нахождения наибольшего общего делителя (НОД) двух натуральных чисел (алгоритм Евклида).

Алгоритм может быть следующим:

- 1) Задать два числа.
- 2) Если числа равны, то взять любое из них в качестве ответа и остановиться, в противном случае продолжить выполнение алгоритма.
- 3) Определить большее из чисел.
- 4) Заменить большее из чисел разностью большего и меньшего из чисел.
- 5) Повторить алгоритм с шага 2.

Описанный алгоритм применим к любым натуральным числам и должен приводить к решению поставленной задачи.

Графический способ

При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий. Такое графическое представление называется *схемой алгоритма*,

или *блок-схемой*. В блок-схеме каждому типу действий соответствует геометрическая фигура, представленная в виде блочного символа. В таблице приведены наиболее часто употребляемые символы.

Название	Блок-схема	Пояснение
Пуск-останов		Начало, конец алгоритма, вход и выход в подпрограмму
Процесс		Вычислительное действие или последовательность действий
Решение		Проверка условий
Модификация		Начало цикла
Предопределённый процесс		Вычисления по подпрограмме
Ввод-вывод		Ввод-вывод в общем виде

Блок «процесс» применяется для обозначения действия или последовательности действий, изменяющих значение, форму представления или размещения данных. Для улучшения наглядности схемы несколько отдельных блоков обработки можно объединять в один блок. Представление отдельных операций достаточно свободно.

Блок «решение» используется для обозначения переходов управления по условию. В каждом блоке «решение» должны быть указаны вопрос, условие или сравнение, которые он определяет.

Блок «модификация» используется для организации циклических конструкций. Внутри блока записывается параметр цикла, для которого указываются его начальное значение, граничное условие и шаг изменения значения параметра для каждого повторения.

Блок «предопределённый процесс» используется для указания обращений к вспомогательным алгоритмам, существующим автономно в виде некоторых самостоятельных модулей, и для обращений к библиотечным подпрограммам.

Псевдокод

Псевдокод представляет собой систему обозначений и правил, предназначенную для единообразной записи алгоритмов.

В псевдокоде не приняты строгие синтаксические правила для записи команд, присущие формальным языкам, что облегчает запись алгоритма на стадии его проектирования. Однако в псевдокоде обычно имеются некоторые конструкции, присущие формальным языкам. В псевдокоде, также как и в формальных языках, есть служебные слова, смысл которых однозначно определён. Например, алгоритмы на алгоритмическом языке записываются с помощью служебных слов, представленных в таблице I.8.

Таблица I.8. Служебные слова алгоритмического языка

алг (алгоритм)	сим (символьный)	дано	да	нет
арг (аргумент)	лит (литерный)	надо	для	при
рез (результат)	лог (логический)	если	от	до
нач (начало)	таб (таблица)	то	знач	выбор
кон (конец)	нц (начало цикла)	иначе	и	или
цел (целый)	кц (конец цикла)	все	ввод	вывод
вещ (вещественный)	длин (длина)	пока	утв	не

Общий вид алгоритма:

алг название алгоритма (аргументы и результаты)

дано условия применимости алгоритма

надо цель выполнения алгоритма

нач описание промежуточных величин

последовательность команд (тело алгоритма)

кон

Часть алгоритма от слова **алг** до слова **нач** называется заголовком, а часть, заключённая между словами **нач** и **кон**, — телом алгоритма.

Программный способ записи алгоритмов

Алгоритм, предназначенный для исполнения на компьютере, должен быть записан на понятном ему языке. В этом случае язык для записи алгоритмов должен быть формализован. Такой язык принято называть *языком программирования*, а запись алгоритма на этом языке — *программой*.

5.2. Основные алгоритмические конструкции

1. Структура **следование**. Образуется последовательностью действий, следующих одно за другим:

Алгоритмический язык	Блок-схема
действие 1 действие 2 ... действие n	<pre> graph TD Start(()) --> B1[действие 1] B1 --> B2[действие 2] B2 --> Dots[...] Dots --> Bn[действие n] Bn --> End(()) </pre>

2. Структура **ветвление**. В зависимости от результата проверки условия («да» или «нет») осуществляет выбор одного из альтернативных путей работы алгоритма. Каждый из путей ведёт к общему выходу, поэтому работа алгоритма будет продолжаться независимо от того, какой путь будет выбран. Структура «ветвление» бывает четырёх видов: «если-то»; «если-то-иначе»; «выбор»; «выбор-иначе».

Структура «если-то»

Алгоритмический язык	Блок-схема
если условие то действия все	<pre> graph TD Start(()) --> Cond{Условие} Cond -- Да --> B[действия] B --> Join(()) Cond -- Нет --> Join Join --> End(()) </pre>

Структура «если-то-иначе»

Алгоритмический язык	Блок-схема
если условие то действия 1 иначе действия 2 все	<pre> graph TD Start(()) --> Cond{Условие} Cond -- Да --> B1[действия 1] Cond -- Нет --> B2[действия 2] B1 --> Join(()) B2 --> Join Join --> End(()) </pre>

Структура «выбор»

Алгоритмический язык	Блок-схема
выбор при условии 1: действия 1 при условии 2: действия 2 ... при условии n : действия n все	<pre> graph TD Start(()) --> D1{Условие 1} D1 -- Да --> P1[действия 1] D1 -- Нет --> D2{Условие 2} D2 -- Да --> P2[действия 2] D2 -- Нет --> Dots[...] Dots --> Dn{Условие n} Dn -- Да --> Pn[действия n] Dn -- Нет --> Exit(()) P1 --> Join(()) P2 --> Join Pn --> Join Join --> Exit </pre>

Структура «выбор-иначе»

Алгоритмический язык	Блок-схема
выбор при условии 1: действия 1 при условии 2: действия 2 ... при условии n : действия n иначе действия $n + 1$ все	<pre> graph TD Start(()) --> D1{Условие 1} D1 -- Да --> P1[действия 1] D1 -- Нет --> D2{Условие 2} D2 -- Да --> P2[действия 2] D2 -- Нет --> Dots[...] Dots --> Dn{Условие n} Dn -- Да --> Pn[действия n] Dn -- Нет --> Pn1[действия n+1] P1 --> Join(()) P2 --> Join Pn --> Join Pn1 --> Join Join --> Exit(()) </pre>

3. Структура **цикл**. Обеспечивает многократное выполнение некоторой совокупности действий, которая называется телом цикла. Циклы бывают трёх видов: с предусловием «пока-делай», с постусловием «делай-пока», со счётчиком «для».

Цикл с предусловием («пока-делай»).

Предписывает выполнять тело цикла до тех пор, пока выполняется условие, записанное после слова **пока**.

Алгоритмический язык	Блок-схема
нц пока условие тело цикла кц	

Цикл с постусловием («делай-пока»).

Предписывает выполнять тело цикла до тех пор, пока выполняется условие, записанное после слова **пока**. В отличие от цикла «пока-делай», тело цикла выполняется хотя бы один раз.

Алгоритмический язык	Блок-схема
нц тело цикла кц_при условие	

Цикл со счётчиком («для»).

Предписывает выполнять тело цикла для всех значений некоторой переменной (параметра цикла) в заданном диапазоне.

Алгоритмический язык	Блок-схема
нц для i от k до m тело цикла кц	

§ 6. Файловые системы

6.1. Основные понятия

Файл (англ. *file* — папка) — это именованная совокупность любых данных, размещённая на внешнем запоминающем устройстве и хранимая, пересылаемая и обрабатываемая как единое целое. Файл может содержать программу, числовые данные, текст, закодированное изображение и др.

Файловая система — это средство для организации хранения файлов на каком-либо носителе.

Имя файла состоит из двух частей, разделённых точкой: собственно имя файла и расширение, определяющее его тип (программа ли это, данные и т. д.). Имя файлу даёт пользователь, а тип файла обычно задаётся программой автоматически при его создании.

Некоторые типы файлов и расширений

Тип файла	Расширения
Исполняемые файлы	.exe, .com
Текстовые файлы	.txt, .ert, .doc
Графические файлы	.jpg, .bmp, .gif, .tiff
Аудиофайлы	.wav, .mid, .mp3
Видеофайлы	.avi, .mpg, .mpeg, .wmv
Интернет-страницы	.htm, .html
Архивные файлы	.rar, .zip

Путь к файлу. В путь к файлу входят записываемые через разделитель «\» логическое имя диска и последовательность имён вложенных друг в друга каталогов, в последнем из которых содержится нужный файл.

Путь к файлу вместе с именем файла называют *полным именем файла*.

Пример полного имени файла:

C:\My_doc\Lesson\lesson1.doc

6.2. Маски имён файлов

Маски имён файлов используются для групповых операций с файлами. Для задания масок имён файлов используют следующие символы:

«*» — набор произвольных символов (возможно, и ни одного);

«?» — один произвольный символ.

Пример 6.1. По маске *. * будут отобраны абсолютно все файлы.

Пример 6.2. По маске *.doc будут отобраны все файлы с расширением doc.

Пример 6.3. По маске `a?*.*.txt` будут отображены файлы, в имени которых на первом месте стоит символ «a», на третьем — «с», а между ними расположен ровно один символ. Остальные символы могут быть произвольными.

Пример 6.4. Предположим, что в каталоге есть следующие файлы:

<code>My1.doc</code>	<code>My321.dcc</code>
<code>My2.doc</code>	<code>My34.dc</code>

По маске `My*.d*` будут отображены все четыре файла. По маске `M*1.d?c` будут отображены файлы `My1.doc` и `My321.dcc`. По маске `M?2*.d?` не будет отображено ни одного файла.

§ 7. Компьютерные сети

7.1. Основные понятия

Компьютерная сеть (англ. *Net*) — это совокупность ЭВМ и других устройств, соединённых линиями связи и обменивающихся информацией между собой в соответствии с определёнными правилами — протоколами.

Локальная сеть объединяет несколько компьютеров и даёт возможность пользователям совместно использовать ресурсы этих компьютеров, а также подключённых к сети периферийных устройств (принтеров, плоттеров, дисков, модемов и др.).

Под **протоколом передачи данных** подразумевается набор соглашений, который обеспечивает обмен данными между компьютерами. Протоколы задают способы передачи данных и обработки ошибок в сети независимо от аппаратной платформы.

Протокол IP (Internet Protocol) — протокол межсетевого взаимодействия, отвечающий за адресацию и позволяющий пакету на пути к конечному пункту назначения проходить по нескольким сетям.

Протокол TCP (Transmission Control Protocol) — протокол управления передачей данных с установкой соединения между двумя узлами сети, использующий автоматическую повторную передачу потерянных или содержащих ошибки пакетов и устраняющий возможность их дублирования. Этот протокол отвечает за разбиение передаваемой информации на пакеты и правильное восстановление информации из пакетов получателем.

Поисковый сервер (поисковая машина) — специальный сервер, оказывающий услуги по поиску информации в Интернете. На поисковом сервере можно найти адрес сайта, который содержит нужную информацию.

Единица измерения скорости передачи данных — бит в секунду.

7.2. Адресация в сети

Каждый компьютер, подключённый к сети Интернет, может иметь два равноценных уникальных адреса: цифровой IP-адрес и символическое доменное имя.

IP-адрес компьютера имеет длину 4 байта. Обычно первый и второй байты определяют адрес сети, третий байт определяет адрес подсети, а четвёртый — адрес компьютера в подсети. Для удобства IP-адрес записывают в виде четырёх чисел со значениями от 0 до 255, разделённых точками, например: 125.57.4.120. Здесь адрес сети — 125.57; адрес подсети — 4; адрес компьютера в подсети — 120.

7.3. Маска сети

В терминологии сетей TCP/IP **маской сети** или **маской подсети** (network mask) называется битовая маска (bitmask), определяющая, какая часть IP-адреса (ip address) узла (host) сети относится к адресу сети, а какая — к адресу самого узла в этой сети.

Под маску отводится 32-бита. В двоичном виде в маске подсети сначала идёт N единиц, потом $32 - N$ нулей. N называют *длиной маски* (адрес подсети), $32 - N$ нулей — *идентификатором хоста* внутри подсети. Например,

$255.255.254.0 = 11111111.11111111.11111110.00000000$.

Здесь длина маски $N = 23$.

Маски подсети являются основой метода бесклассовой маршрутизации (CIDR). При этом подходе маску подсети записывают вместе с IP-адресом в формате «IP-адрес/количество единичных бит в маске». Число после знака дроби означает количество единичных разрядов в маске подсети. Например, записи

192.168.11.10/255.255.254.0 соответствует

192.168.1.12/23.

Чтобы получить адрес подсети, зная IP-адрес и маску подсети, необходимо применить к ним операцию поразрядной конъюнкции (логическое И).

Пример 7.1. По заданным IP-адресу узла и маске 192.168.11.10/23 определите адрес сети.

Решение. Осуществив побитовое умножение IP-адреса узла и маски, получим:

IP-адрес	
11000000.10101000.00001011.00001010	192.168.11.10
Маска подсети	
11111111.11111111.11111110.00000000	255.255.254.0
Адрес сети	
11000000.10101000.00001010.00000000	192.168.10.0

Ответ. 192.168.10.0

Прежде чем посылать IP-пакет, компьютер определяет, попадает ли адрес назначения в «свою» подсеть. Если попадает, то компьютер шлёт пакет «напрямую», если же нет — отправляет его шлюзу по умолчанию (маршрутизатору).

Количество хостов в подсети определяется по формуле: $2^{32 - N} - 2$, где N — длина маски. Чем длиннее маска, тем меньше в ней хостов.

Глава II

Учебно-тренировочные тесты

Инструкция по выполнению работы

На выполнение экзаменационной работы по информатике и ИКТ отводится 235 минут. Экзаменационная работа состоит из двух частей, включающих 27 задания. На выполнение заданий части 1 рекомендуется отводить 1,5 часа (90 минут). Остальное время рекомендуется отводить на выполнение заданий части 2.

Часть 1 содержит 23 задания с кратким ответом. В экзаменационной работе предложены следующие разновидности заданий с кратким ответом:

- задания на выбор и запись одного или нескольких правильных ответов из предложенного перечня ответов;
- задания на вычисление определенной величины;
- задания на установление правильной последовательности, представленной в виде строки символов по определенному алгоритму.

Часть 2 состоит из 4 заданий. Для выполнения заданий этой части вам необходимо написать развернутый ответ в произвольной форме.

Выполняйте задания в том порядке, в котором они даны. Если какое-то задание вызывает у вас затруднение, пропустите его и переходите к следующему. К пропущенным заданиям можно будет вернуться, если останется время.

За каждый правильный ответ в зависимости от сложности задания даётся один или более баллов. Каждое задание части 1 оценивается в 1 балл. Задания части 2 оцениваются следующим образом: задания 24, 26 — 3 балла, задание 25 — 2 балла, задание 27 — 4 балла. Баллы, полученные вами за все выполненные задания, суммируются. Наибольшее количество первичных баллов, которое можно получить за верное выполнение всех заданий — 35. Постарайтесь выполнить как можно большее количество заданий и набрать как можно больше баллов.

Желаем успеха!

§ 1. Учебно-тренировочные тесты

Вариант № 1

Часть 1

1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, используется неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Вот этот код: А — 01, Б — 10, В — 110, Г — 001, Д — 111. Можно ли сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было однозначно декодировать? Коды остальных букв меняться не должны. Выберите правильный вариант ответа.

- | | |
|---------------------|---------------------|
| 1) это невозможно | 2) для буквы В — 11 |
| 3) для буквы Г — 00 | 4) для буквы Д — 11 |

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	F
			1		0			0
		1				1		1
0				1				1

Каким выражением может быть F ?

- 1) $x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge x_4 \wedge x_5 \wedge \neg x_6 \wedge \neg x_7 \wedge x_8$
- 2) $x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4 \vee x_5 \vee \neg x_6 \vee \neg x_7 \vee x_8$
- 3) $\neg x_1 \vee x_2 \vee x_3 \vee \neg x_4 \vee \neg x_5 \vee x_6 \vee x_7 \vee \neg x_8$
- 4) $\neg x_1 \wedge x_2 \wedge x_3 \wedge \neg x_4 \wedge \neg x_5 \wedge x_6 \wedge x_7 \wedge \neg x_8$

3. Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы: символ «?» (вопросительный знак) означает ровно один произвольный символ; символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

В каталоге находятся шесть файлов:

dextrose.txt

expect.bst

flexible.tex

deluxe.bst

luxе.txt

subexchange.html

Определите, по какой из масок из них будет отображена указанная группа файлов:

dextrose.txt

flexible.tex

1) ?ex*.t??

2) *ex?.tx*

3) ?ex*.t??

4) *ex*.t??

4. Сколько единиц содержится в двоичной записи восьмеричного числа 125_8 ?

5. Между населёнными пунктами A, B, C, D, E, F построены дороги, протяжённость которых приведена в таблице. (Прочерк в таблице означает, что прямой дороги между соответствующими пунктами нет.)

	A	B	C	D	E	F
A	-	4	3	10	-	20
B	4	-	-	4	-	-
C	3	-	-	6	-	-
D	10	4	6	-	5	11
E	-	-	-	5	-	8
F	20	-	-	11	8	-

Определите длину кратчайшего пути между пунктами A и F (при условии, что передвигаться можно только по построенным дорогам).

6. Автомат получает на вход трёхзначное десятичное число. По полученному числу строится новое десятичное число по следующим правилам:

1) вычисляются два числа — сумма старшего и среднего разрядов, а также сумма среднего и младшего разрядов заданного числа;

2) полученные два числа записываются друг за другом в порядке неубывания (без разделителей).

Пример. Исходное число: 326. Поразрядные суммы: 5, 8. Результат: 58.

Укажите наименьшее число, в результате обработки которого автомат выдаст число 1012.

7. Пете нужно с помощью электронных таблиц построить таблицу сложения для значений x и y от 13 до 16. Для этого сначала в диапазонах B1:E1 и A2:A5 он записал числа от 13 до 16. Затем в ячейку E2 записал формулу сложения, после чего скопировал её во все ячейки диапазона B2:E5. В итоге на экране получился фрагмент таблицы:

	A	B	C	D	E
1		13	14	15	16
2	13	26	27	28	29
3	14	27	28	29	30
4	15	28	29	30	31
5	16	29	30	31	32

Какая формула была записана в ячейке E2?

- 1) = \$A2 + E\$1 2) = E\$1 + A\$2
 3) = \$A2 + \$E1 4) = A2 + E\$1

8. Определите, что будет напечатано в результате выполнения программы, записанной ниже на различных языках программирования.

Бейсик	Алгоритмический язык
<pre> K = 9 S = 43 WHILE S >= 0 K = K - 1 S = S - 2 * K WEND PRINT K </pre>	<pre> алг нач цел k, s k := 9; s := 43 нц пока s >= 0 k := k - 1; s := s - 2 * k кц вывод k кон </pre>
Паскаль	Си
<pre> var k, s: integer; begin k := 9; s := 43; while s >= 0 do begin k := k - 1; s := s - 2 * k end; writeln(k); end. </pre>	<pre> #include <stdio.h> void main() { int k, s; k = 9; s = 43; while (s >= 0) { k -= 1; s -= 2 * k; } printf("%d", k); } </pre>

9. Производится одноканальная (моно) звукозапись с частотой дискретизации 16 кГц и 32-битным разрешением, результаты записываются в файл, сжатие данных не используется. В результате записи был получен файл размером 6 Мбайт. Определите приблизительно, сколько времени (в секундах) производилась запись. В качестве ответа укажите ближайшее к времени записи целое число, кратное 10.

10. Все четырёхбуквенные слова, составленные из букв А, В, О, С, записаны в алфавитном порядке. Вот начало списка:

1. АААА
2. АААВ
3. АААО
4. АААС
5. ААВА ...

Запишите, на каком месте от начала списка стоит слово СОВА.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> SUB F(n) PRINT n, ' ' IF n > 1 THEN F(n - 1) F(n - 2) END IF END SUB </pre>	<pre> алг F(цел n) нач вывод n, ' ' если n > 1 то F(n - 1) F(n - 2) все кон </pre>
Паскаль	Си
<pre> procedure F(n: integer); begin write(n, ' '); if n > 1 then begin F(n - 1); F(n - 2); end end; </pre>	<pre> void F(int n) { printf("%d ", n); if (n > 1) { F(n - 1); F(n - 2); } } </pre>

Чему равна сумма чисел, напечатанных на экране при выполнении вызова F(5)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 139.95.201.73

Маска: 255.255.248.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
139	201	208	255	95	200	0	25

Пример. Пусть искомый IP-адрес 192.117.182.0, и дана таблица

A	B	C	D	E	F	G	H
182	117	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Было выпущено 2000 лотерейных билетов, пронумерованных от 1 до 2000. При продаже билета в специальное устройство заносится его номер с использованием минимально возможного количества бит, одинакового для каждого билета. Определите объём памяти (в байтах), отведённый устройством, после того как было продано 40 билетов. В ответе запишите только число, слово «байт» писать не нужно.

14. Исполнитель *Чертёжник* перемещается на координатной плоскости, оставляя след в виде линии. *Чертёжник* может выполнять команду **Сместиться на** (a, b) (где a, b — целые числа), перемещающую *Чертёжника* из точки с координатами (x, y) в точку с координатами $(x + a, y + b)$. Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные — уменьшается. Например, если *Чертёжник* находится в точке с координатами $(2, 3)$, то команда **Сместиться на** $(-5, 2)$ переместит *Чертёжника* в точку $(-3, 5)$.

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что *последовательность команд* будет выполнена указанное число раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами n , a , b обозначены неизвестные числа, при этом $n > 1$):

НАЧАЛО

ПОВТОРИ n РАЗ

Сместиться на $(-24, -7)$

Сместиться на (a, b)

Сместиться на $(15, 23)$

КОНЕЦ ПОВТОРИ

Сместиться на $(9, 16)$

КОНЕЦ

Укажите наименьшее возможное значение числа n , для которого найдутся такие значения чисел a и b , что Чертёжник, начав выполнение программы из точки с координатами $(-18, 14)$, после выполнения алгоритма окажется в точке $(12, 15)$.

15. На рисунке 1 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

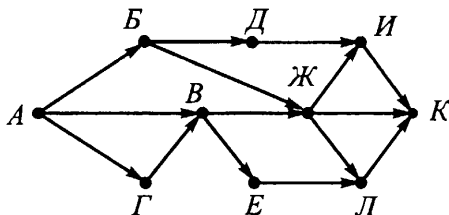


Рис. 1.

16. Сколько значащих нулей содержится в двоичной записи числа, которое можно представить в виде $8^{740} - 2^{900} + 7$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Страны	380
Испания	350
Курорты	440
Страны Курорты	820
Страны & Испания	85
Курорты & Испания	85

Сколько страниц (в сотнях тысяч) будет найдено по запросу
(Страны | Курорты) & Испания?

18. На числовой прямой даны два отрезка: $P = [18, 27]$ и $Q = [20, 54]$. Укажите наименьшую возможную длину такого отрезка A , что логическое выражение

$$\neg(x \in P) \rightarrow ((x \in Q) \vee (x \in P)) \rightarrow (x \in A)$$

тождественно истинно, то есть принимает значение 1 при любом значении переменной x .

19. В программе обрабатывается одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8; 5; 11; 4; 6; 9; 4; 10; 7; 8 соответственно, т.е. $A[0] = 8$; $A[1] = 5$ и т.д.

Определите значение переменной s после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> k = 0 t = 0 FOR i = 0 TO 9 IF A(i) > k THEN t = k: k = A(i) ELSE IF A(i) > t THEN t = A(i) END IF END IF NEXT i s = k + t </pre>	<pre> k:=0 t:=0 нц для i от 0 до 9 если A[i]>k то t:=k; k:=A[i] иначе если A[i]>t то t:=A[i] все все кц s:=k+t </pre>

Паскаль	Си
<pre> k:=0; t:=0; for i:=0 to 9 do if A[i]>k then begin t:=k; k:=A[i] end else t:=A[i]; s:=k+t; </pre>	<pre> k=0; t=0; for (i=0; i<=9; i++) do if (A[i]>k) { t=k; k=A[i]; } else t=A[i]; s=k+t; </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наибольшее из таких чисел x , при вводе которого алгоритм печатает сначала 3, а потом 21.

Бейсик	Паскаль
<pre> INPUT X A = 0 : B = 0 WHILE X > 0 A = A + 1 B = B + X MOD 10 X = X \ 10 WEND B = 45 - B PRINT A PRINT B </pre>	<pre> var x, a, b : integer; begin readln(x); a := 0; b := 0; while x > 0 do begin a := a + 1; b := b + x mod 10; x := x div 10 end; b := 45 - b; writeln(a); writeln(b); end. </pre>
Алгоритмический язык	Си
<pre> алг нач цел x, a, b ввод x a := 0; b := 0 нц пока x > 0 a := a + 1 b := b + mod(x, 10) x := div(x, 10) кц b := 45 - b вывод a, нс, b кон </pre>	<pre> #include <stdio.h> void main() { int x, a, b; scanf("%d", &x); a = 0; b = 0; while (x > 0) { a += 1; b += (x % 10); x /= 10; } b = 45 - b; printf("%d \n %d", a, b); } </pre>

21. Ниже на четырёх языках записан алгоритм. Получив на вход целое число k , этот алгоритм печатает число i . Укажите число различных значений входной переменной k , при которых программа выдаёт тот же ответ, что и при входном значении $k = 80$. Значение $k = 80$ также включается в подсчёт различных значений k .

Бейсик	Алгоритмический язык
<pre> INPUT K I = 1 WHILE I<40 AND F(I)<=K I = 2 * I WEND PRINT I FUNCTION F (X) F = 5*X END FUNCTION </pre>	<pre> алг нач цел i,k ввод k i:=1 нц пока i<40 и f(i)<=k i:=2*i кц вывод i кон алг цел F(цел x) нач знач := 5 * x кон </pre>
Паскаль	Си
<pre> var k,i: integer; Function F(x:integer):integer; begin F:=5*x end; begin readln(k); i:=1; while (i<40) and (f(i)<=k) do i:=2*i; writeln(i) end. </pre>	<pre> #include<stdio.h> int F(int x) { return 5*x; } void main() { int k, i; scanf("%d", &k); i=1; while (i<40 && F(i)<=k) i*=2; printf("%d", i); } </pre>

22. У исполнителя X164 три команды, которым присвоены номера:

1. прибавь 1,
2. прибавь 6,
3. умножь на 4.

Первая из них увеличивает число на экране на 1, вторая — на 6, а третья — в 4 раза. Программа для исполнителя X164 — это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 20?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \vee \neg x_2) \wedge (x_3 \rightarrow (\neg x_1 \wedge x_2)) \wedge (\neg x_1 \vee y_1) = 1,$$

$$(x_2 \vee \neg x_3) \wedge (x_4 \rightarrow (\neg x_2 \wedge x_3)) \wedge (\neg x_2 \vee y_2) = 1,$$

...

$$(x_6 \vee \neg x_7) \wedge (x_8 \rightarrow (\neg x_6 \wedge x_7)) \wedge (\neg x_6 \vee y_6) = 1,$$

$$(x_7 \vee \neg x_8) \wedge (\neg x_7 \vee y_7) = 1,$$

$$(\neg x_8 \vee y_8) = 1.$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры поступает последовательность из пяти неотрицательных целых чисел (некоторые числа могут быть одинаковыми). Программа должна вывести на экран сумму чисел, кратных трём, в исходной последовательности и минимальное из чисел, кратных трём. Если чисел, кратных трём, нет, требуется на экран вывести «NO». Известно, что вводимые числа не превышают 1000. Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на четырёх языках программирования.

Алгоритмический язык

```
алг
нач
  цел N=5, i, X, MinX, S
  MinX:=3
  S:=0
  нц для i от 1 до N
    ввод X
    если mod(X, 3) = 0 то
      S:=S+X;
      если X < MinX то
        MinX:=i;
      все
    все
  кц
  если S > 0 то
    вывод S, нс, MinX
  иначе
    вывод "No"
  все
кон
```

Паскаль

```
const N=5;
var i, X, MinX, S: integer;
begin
  MinX:=3;
  S:=0;
  for i := 1 to N do begin
    readln(X);
    if X mod 3 = 0 then begin
      S:= S + X;
      if x < MinX then
        MinX:= i
      end
    end;
    if S > 0 then begin
      writeln(S);
      writeln(MinX)
    end
    else writeln('No')
  end.
end.
```

Бейсик

```
CONST N = 5
S = 0
MinX = 3
FOR I = 1 TO N
    INPUT X
    IF X mod 3 = 0 THEN
        S = S + X
        IF X < MinX THEN
            MinX = i
        END IF
    END IF
NEXT I
IF S > 0 THEN
    PRINT S
    PRINT MinX
ELSE
    PRINT "No"
END IF
END
```

Си

```
#include <stdio.h>
void main() {
    const int N = 5;
    int i, X, MinX, S;
    MinX=3;
    S=0;
    for (i = 1; i <= N; i++) {
        scanf("%d ", &X);
        if (X % 3 == 0) {
            S+= X;
            if (X < MinX)
                MinX= i;
        }
    }
    if (S > 0) {
        printf("%d\n %d\n", S, MinX);
    }
    else printf("No")
}
```

Последовательно выполните следующее:

1. Напишите, что выведет эта программа при вводе последовательности чисел 5 6 9 7 12.

2. Приведите пример такой последовательности, содержащей хотя бы одно число, кратное трём, что, несмотря на ошибки, программа печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

1) выпишите строку, в которой сделана ошибка;

2) укажите, как исправить ошибку, — приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от -20 до 20 , отличные от 0 . Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести наибольшее количество идущих подряд элементов одного знака. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

Бейсик	Алгоритмический язык
CONST N = 30	алг
DIM A(N) AS INTEGER	нач
DIM I AS INTEGER	цел N=30
DIM K AS INTEGER	целтаб а[1:N]
DIM MAX AS INTEGER	цел i, k, max
FOR I = 1 TO N	нц для i от 1 до N
INPUT A(I)	ввод а[i]
NEXT I	кц
...	...
END	кон

Паскаль	Си
<pre>const N=30 var a: array [1..N] of integer; i, k, max: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>#define N 30 void main(void){ int a[N]; int i, k, max; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>

В качестве ответа Вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

26. Два игрока, Коля и Саша, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Коля. За один ход игрок может добавить в кучу один или два камня или увеличить количество камней в куче в два раза. Например, имея кучу из 15 камней, за один ход можно получить кучу из 16, 17 или 30 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней. Игра завершается в тот момент, когда количество камней в куче становится не менее 23. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 23 или больше камней. В начальный момент в куче было S камней, $1 \leq S \leq 22$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

1. а) Укажите все такие значения числа S , при которых Коля может выиграть в один ход. Обоснуйте, что найдены все нужные значения S , и укажите выигрышный ход для каждого указанного значения S .

б) Укажите такое значение S , при котором Коля не может выиграть за один ход, но при любом ходе Коли Саша может выиграть своим первым ходом. Опишите выигрышную стратегию Саши.

2. Укажите два таких значения S , при которых у Коли есть выигрышная стратегия, причём

- Коля не может выиграть за один ход, и
- Коля может выиграть своим вторым ходом независимо от того, как будет ходить Саша. Для каждого указанного значения S опишите выигрышную стратегию Коли.

3. Укажите значение S , при котором

- у Саши есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Коли, и
- у Саши нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения S опишите выигрышную стратегию Саши. Постройте дерево всех партий, возможных при этой выигрышной стратегии Саши (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход, в узлах — количество камней в куче.

27. У Димы есть много книг, которые он ещё не прочитал. Дима обожает толстые старые книги. Кроме того, он не любит произведения с длинными названиями. В очередной раз, когда ему надо было выбрать себе книгу, он решил воспользоваться помощью компьютера. Мальчик составил список непрочитанных книг и определил критерии, по которым необходимо выбрать книгу: год издания должен быть ранее 1980, количество страниц — не менее 300, а название должно быть по возможности самым коротким из названий книг, удовлетворяющих первым двум условиям. Гарантирует-ся, что хотя бы одна книга удовлетворяет перечисленным критериям.

Напишите эффективную по времени работы и по используемой памяти программу, которая по имеющемуся каталогу непрочитанных книг определяет количество толстых старых книг, а из них выбирает обладающую самым коротким названием. На вход программе сначала подаётся число имеющихся у Димы книг N . В каждой из следующих N строк находится описание какой-либо книги в следующем формате:

<Фамилия автора> <Год издания> <Кол-во страниц> <Название>, где <Год издания>, <Кол-во страниц> — целые числа;

<Фамилия автора> — строка без пробелов, состоящая не более чем из 20 символов;

<Название> — строка, состоящая не более чем из 40 символов.

<Фамилия автора>, <Год издания>, <Кол-во страниц>, <Название> разделены между собой одним пробелом.

Пример входной строки:

Казанцев 1988 637 Клокоцущая пустота

Программа должна выводить количество книг в списке, изданных ранее 1980 года и содержащих не менее 300 страниц, а также наименование книги с самым коротким названием.

Вариант № 2

Часть 1

1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, используется неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Вот этот код: А — 01, Б — 101, В — 110, Г — 001, Д — 111.

Можно ли сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было декодировать однозначно? Коды остальных букв меняться не должны. Выберите правильный вариант ответа.

1) это невозможно

2) для буквы Б — 10

3) для буквы В — 10

4) для буквы Г — 0

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	F
		0				0		1
0			0					1
	0				0			0

Каким выражением может быть F ?

1) $x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge x_4 \wedge x_5 \wedge \neg x_6 \wedge \neg x_7 \wedge \neg x_8$

2) $\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4 \vee x_5 \vee \neg x_6 \vee \neg x_7 \vee x_8$

3) $\neg x_1 \vee x_2 \vee \neg x_3 \vee \neg x_4 \vee \neg x_5 \vee x_6 \vee x_7 \vee \neg x_8$

4) $\neg x_1 \wedge x_2 \wedge x_3 \wedge \neg x_4 \wedge \neg x_5 \wedge \neg x_6 \wedge x_7 \wedge \neg x_8$

3. Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы: символ «?» (вопросительный знак) означает ровно один произвольный символ; символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

В каталоге находятся шесть файлов:

athlete.txt	satchel.bst
thought.bst	path.txt
rethink.tlog	other.thmx

Определите, по какой из масок из них будет отобрана указанная группа файлов:

athlete.txt
thought.bst
path.txt

1) ?th*.?t? 2) *th*.??t 3) *th?.t* 4) ?th*.?t

4. Сколько единиц содержится в двоичной записи восьмеричного числа 703_8 ?

5. Между населёнными пунктами A, B, C, D, E, F построены дороги, протяжённость которых приведена в таблице. (Прочерк в таблице означает, что прямой дороги между соответствующими пунктами нет.)

	A	B	C	D	E	F
A	-	3	6	-	7	14
B	3	-	4	-	-	-
C	6	4	-	2	-	-
D	-	-	2	-	-	8
E	7	-	-	-	-	5
F	14	-	-	8	5	-

Определите длину кратчайшего пути между пунктами A и F (при условии, что передвигаться можно только по построенным дорогам).

6. Автомат получает на вход четырёхзначное число. Из этого числа строится новое число по следующим правилам.

1. Вычисляются три числа — сумма первой и второй цифр, сумма второй и третьей цифр и сумма третьей и четвёртой цифр.

2. Полученные три числа записываются друг за другом в порядке возрастания (без разделителей).

Пример. Исходное четырёхзначное число: 7348. Сумма первой и второй цифр — 10. Сумма второй и третьей цифр — 7. Сумма третьей и четвёртой цифр — 12. Результат: 71012.

Укажите наименьшее число, в результате обработки которого автомат выдаст число 81116.

7. Саше нужно с помощью электронных таблиц построить таблицу значений формулы $x - 3y$ для значений x и y от 2 до 5. Для этого сначала в диапазонах B1:E1 и A2:A5 он записал числа от 2 до 5. Затем в ячейку B2 записал формулу (A2 — значение x , B1 — значение y), после чего скопировал её во все ячейки диапазона B2:E5. В итоге на экране получился фрагмент таблицы:

	A	B	C	D	E
1		2	3	4	5
2	2	-4	-7	-10	-13
3	3	-3	-6	-9	-12
4	4	-2	-5	-8	-11
5	5	-1	-4	-7	-10

Какая формула была записана в ячейке B2?

1) = A\$2 - 3 * B\$1

2) = \$A2 - 3 * B\$1

3) = A\$2 - 3 * \$B1

4) = \$A2 - 3 * \$B1

8. Определите, что будет напечатано в результате выполнения программы, записанной ниже на различных языках программирования.

Паскаль	Си
<pre> var k, s: integer; begin k := 5; s := 4; while s >= 0 do begin k := k - 1; s := s + k; end; writeln(k); end.</pre>	<pre> #include <stdio.h> void main() { int k, s; k = 5; s = 4; while (s >= 0) { k -= 1; s += k; } printf("%d", k); }</pre>

Бейсик	Алгоритмический язык
<pre> K = 5 S = 4 WHILE S >= 0 K = K - 1 S = S + K WEND PRINT K </pre>	<pre> алг нач цел k, s k := 5; s := 4 нц пока s >= 0 k := k - 1; s := s + k кц вывод k кон </pre>

9. Производится двухканальная (стерео) звукозапись с частотой дискретизации 48 кГц и 24-битным разрешением, результаты записываются в файл, сжатие данных не используется. В результате записи был получен файл размером 80 Мбайт. Определите приблизительно, сколько времени (в секундах) производилась запись. В качестве ответа укажите ближайшее к времени записи целое число, кратное 10.

10. Все четырёхбуквенные слова, составленные из букв А, Д, К, Л, записаны в алфавитном порядке. Вот начало списка:

1. АААА
2. АААД
3. АААК
4. АААЛ
5. ААДА ...

Запишите, на каком месте от начала списка стоит слово КЛАД.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> SUB F(n) PRINT n, ' ' IF n < 4 THEN F(n + 1) F(2 * n) END IF END SUB </pre>	<pre> алг F(цел n) нач вывод n, ' ' если n < 4 то F(n+1) F(2*n) все кон </pre>

Паскаль	Си
<pre> procedure F(n: integer); begin write(n, ' '); if n < 4 then begin F(n+1); F(2*n) end end; </pre>	<pre> void F(int n) { printf("%d ", n); if (n < 4) { F(n+1); F(2*n); } } </pre>

Чему равна сумма чисел, напечатанных на экране при выполнении вызова $F(1)$?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 219.215.211.57

Маска: 255.255.240.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
0	13	208	215	219	211	240	255

Пример. Пусть искомый IP-адрес 192.117.182.0, и дана таблица

A	B	C	D	E	F	G	H
182	117	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Секретный код состоит из прописных латинских букв (всего используется 8 различных букв) и имеет длину 11 символов. Каждый такой код в компьютерной системе записывается минимально возможным и одинаковым целым количеством байт. При этом все символы кодируются одним и тем же (минимально возможным) количеством бит.

Определите объём памяти (в байтах), необходимый для хранения 120 таких кодов. В ответе запишите только число, слово «байт» писать не нужно.

14. Исполнитель *Чертёжник* перемещается на координатной плоскости, оставляя след в виде линии. *Чертёжник* может выполнять команду **Сместиться на** (a, b) (где a, b — целые числа), перемещающую *Чертёжника* из точки с координатами (x, y) в точку с координатами $(x + a, y + b)$. Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные — уменьшается. Например, если *Чертёжник* находится в точке с координатами $(2, 3)$, то команда **Сместиться на** $(-5, 2)$ переместит *Чертёжника* в точку $(-3, 5)$.

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что последовательность команд будет выполнена указанное число раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами n, a, b обозначены неизвестные числа, при этом $n > 1$):

НАЧАЛО

Сместиться на $(12, -11)$

ПОВТОРИ n РАЗ

Сместиться на (a, b)

Сместиться на $(26, -17)$

Сместиться на $(-18, 21)$

КОНЕЦ ПОВТОРИ

КОНЕЦ

Укажите наименьшее возможное значение числа n , для которого найдутся такие значения чисел a и b , что *Чертёжник*, начав выполнение программы из точки с координатами $(24, -9)$, после выполнения алгоритма окажется в точке $(10, -7)$.

15. На рисунке 2 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

16. Сколько значащих нулей содержится в двоичной записи числа, которое можно представить в виде $4^{1040} - 8^{400} + 15$?

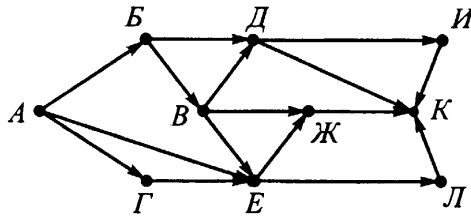


Рис. 2.

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Деревья	560
Плодовые	720
Насекомые	630
(Деревья $ $ Насекомые) $\&$ Плодовые	208
Деревья $\&$ Плодовые	84
Насекомые $\&$ Плодовые	124

Сколько страниц (в сотнях тысяч) будет найдено по запросу
Деревья $|$ Насекомые?

18. На числовой прямой даны два отрезка: $P = [12, 32]$ и $Q = [16, 48]$. Укажите наименьшую возможную длину такого отрезка A , что логическое выражение

$$(x \in P) \rightarrow ((x \in Q) \wedge (x \in P)) \rightarrow \neg(x \in A)$$

тождественно истинно, то есть принимает значение 1 при любом значении переменной x .

19. В программе обрабатывается одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8; 5; 11; 4; 6; 9; 4; 10; 7; 8 соответственно, т.е. $A[0] = 8$; $A[1] = 5$ и т.д.

Определите значение переменной s после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> k = 0: t = 0 FOR i = 0 TO 8 IF A(i) > k THEN k = A(i) : A(i+1):=A(i) ELSE IF A(i) > t THEN t = A(i) END IF END IF NEXT i s = k + t </pre>	<pre> k:=0; t:=0 нц для i от 0 до 8 если A[i]>k то k:=A[i]; A[i+1]:=A[i] иначе если A[i]>t то t:=A[i] все все кц s:=k+t </pre>
Паскаль	Си
<pre> k:=0; t:=0; for i:=0 to 8 do if A[i]>k then begin k:=A[i]; A[i+1]:=A[i] end else t:=A[i]; end s:=k+t; </pre>	<pre> k=0; t=0; for (i=0; i<9; i++) do if (A[i]>k) { k=A[i]; A[i+1]=A[i]; } else t=A[i]; end s=k+t; </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наименьшее из таких чисел x , при вводе которого алгоритм печатает сначала 3, а потом 30.

Бейсик	Паскаль
<pre> INPUT X A = 0 B = 0 WHILE X > 0 A = A + 1 B = B + X MOD 10 X = X \ 10 WEND B = 2*B PRINT A PRINT B </pre>	<pre> var x, a, b : integer; begin readln(x); a := 0; b := 0; while x > 0 do begin a := a + 1; b := b + x mod 10; x := x div 10 end; b := 2*b; writeln(a); writeln(b) end. </pre>

Алгоритмический язык	Си
алг нач цел x, a, b ввод x $a := 0$ $b := 0$ нц пока $x > 0$ $a := a + 1$ $b := b + \text{mod}(x, 10)$ $x := \text{div}(x, 10)$ кц $b := 2 * b$ вывод a, b кон	<pre>#include <stdio.h> void main() { int x, a, b; scanf("%d", &x); a = 0; b = 0; while (x > 0) { a += 1; b += (x % 10); x /= 10; } b = 2*b; printf("%d \n %d", a, b); }</pre>

21. Ниже на четырёх языках записан алгоритм. Получив на вход целое число k , этот алгоритм печатает число i . Укажите число различных значений входной переменной k , при которых программа выдаёт тот же ответ, что и при входном значении $k = 80$. Значение $k = 80$ также включается в подсчёт различных значений k .

Бейсик	Алгоритмический язык
INPUT K $I = 0$ WHILE $I < 20$ AND $F(I) \leq K - I$ $I = 2 + I$ WEND PRINT I FUNCTION $F(X)$ $F = 2 * X$ END FUNCTION	алг нач цел i, k ввод k $i := 0$ нц пока $i < 20$ и $f(i) \leq k - i$ $i := 2 + i$ кц вывод i кон алг цел $F(\text{цел } x)$ нач $\text{знач} := 2 * x$ кон

Паскаль	Си
<pre>var k,i: integer; Function F(x:integer):integer; begin f:=2*x end; begin readln(k); i:=0; while (i<20) and (f(i)<=k-i) do i:=2+i; writeln(i) end.</pre>	<pre>#include<stdio.h> int F(int x) { return 2*x; } void main() { int k, i; scanf("%d", &k); i=0; while (i<20 && F(i)<=k-i) i+=2; printf("%d", i); }</pre>

22. У исполнителя X157 три команды, которым присвоены номера:

1. прибавь 1,
2. прибавь 5,
3. умножь на 7.

Первая из них увеличивает число на экране на 1, вторая — на 5, а третья — в 7 раз. Программа для исполнителя X157 — это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 20?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \wedge \neg x_2) \vee (\neg x_3 \wedge (x_1 \rightarrow x_2)) \vee (x_1 \wedge y_1) = 0,$$

$$(x_2 \wedge \neg x_3) \vee (\neg x_4 \wedge (x_2 \rightarrow x_3)) \vee (x_2 \wedge y_2) = 0,$$

...

$$(x_6 \wedge \neg x_7) \vee (\neg x_8 \wedge (x_6 \rightarrow x_7)) \vee (x_6 \wedge y_6) = 0,$$

$$(x_7 \wedge \neg x_8) \vee (x_7 \wedge y_7) = 0,$$

$$(x_8 \wedge y_8) = 0.$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры поступает последовательность из пяти целых чисел (некоторые числа могут быть одинаковыми).

Программа должна найти в исходной последовательности среднее арифметическое чётных чисел и максимальное чётное число и вывести найденные числа на экран. Если чётных чисел в последовательности нет, требуется на экран вывести «NO». Известно, что вводимые числа больше -1000 и меньше, чем 1000 . Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на четырёх языках программирования.

Алгоритмический язык

```
алг
нач
  цел N=5, i, X, K, Sum, MaxX
  K:=0; Sum:=0
  MaxX:=0
  нц для i от 1 до N
    ввод X
    если mod(X, 2) = 0 то
      K:=K+1
      Sum:=Sum+X
      если X > MaxX то
        MaxX:=X
    все
  все
кц
если Sum <> 0 то
  вывод Sum/N, нс
  вывод MaxX
иначе
  вывод "No"
все
кон
```

Паскаль

```
const N=5;
var i, X, K, Sum, MaxX: integer;
begin
    K:=0; Sum:=0;
    MaxX:=0;
    for i := 1 to N do begin
        readln(X);
        if X mod 2 = 0 then begin
            K:=K+1; Sum:=Sum+X;
            if X > MaxX then MaxX:= X
        end
    end;
    if Sum <> 0 then begin
        writeln(Sum/N:2:1);
        writeln(MaxX)
    end
    else writeln('No')
end.
```

Бейсик

```
CONST N = 5
K:=0: Sum:=0
MaxX = 0
FOR I = 1 TO N
    INPUT X
    IF X mod 2 = 0 THEN
        K = K + 1 : Sum = Sum + X
        IF X < MaxX THEN MaxX = X
    END IF
NEXT I
IF Sum <> 0 THEN
    PRINT Sum/N
    PRINT MaxX
ELSE
    PRINT "No"
END IF
END
```

Си

```
#include <stdio.h>
void main() {
    const int N = 5;
    int i, X, K, Sum, MaxX;
    K=0; Sum=0;
    MaxX=0;
    for (i = 1; i <= N; i++) {
        scanf("%d ", &X);
        if (X % 2 == 0) {
            K=K+1;
            Sum=Sum+X;
            if (X > MaxX)
                MaxX= X;
        }
    }
    if (Sum <> 0) {
        printf("%f\n", (float)Sum/N);
        printf("%d\n", MaxX);
    }
    else printf("No")
}
```

Последовательно выполните следующее:

1. Напишите, что выведет эта программа при вводе последовательности чисел —6 5 — 8 4 2.

2. Приведите пример такой последовательности, содержащей хотя бы одно чётное число, что, несмотря на ошибки, программа печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

1) выпишите строку, в которой сделана ошибка;

2) укажите, как исправить ошибку, — приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования. Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую дру-

гой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от -20 до 20 . Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти суммы идущих подряд (не менее двух) чётных элементов и вывести наибольшую из них.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

Бейсик	Алгоритмический язык
<pre> CONST N = 30 DIM A(1 TO N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM L AS INTEGER DIM MAX AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END </pre>	<pre> алг нач цел N=30 целтаб a[1:N] цел i, k, l, max нц для i от 1 до N ввод a[i] кц ... кон </pre>
Паскаль	Си
<pre> const N=30 var a: array [1..N] of integer; i, k, l, max: integer; begin for i := 1 to N do readln(a[i]); ... end. </pre>	<pre> #define N 30 void main(void){ int a[N]; int i, k, l, max; for (i=0; i<N; i++){ scanf("%d", &a[i]); ... } </pre>

В качестве ответа Вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы.

В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии (например, в образце, записанном на естественном языке).

26. Два игрока, Коля и Саша, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Коля. За один ход игрок может добавить в кучу один или два камня или увеличить количество камней в куче в два раза. Например, имея кучу из 15 камней, за один ход можно получить кучу из 16, 17 или 30 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней. Игра завершается в тот момент, когда количество камней в куче становится не менее 41. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 41 или больше камней. В начальный момент в куче было S камней, $1 \leq S \leq 40$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

1. а) Укажите все такие значения числа S , при которых Коля может выиграть в один ход. Обоснуйте, что найдены все нужные значения S , и укажите выигрышный ход для каждого указанного значения S .

б) Укажите такое значение S , при котором Коля не может выиграть за один ход, но при любом ходе Коли Саша может выиграть своим первым ходом. опишите выигрышную стратегию Саши.

2. Укажите два таких значения S , при которых у Коли есть выигрышная стратегия, причём

— Коля не может выиграть за один ход, и

— Коля может выиграть своим вторым ходом независимо от того, как будет ходить Саша. Для каждого указанного значения S опишите выигрышную стратегию Коли.

3. Укажите значение S , при котором

— у Саши есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Коли, и

— у Саши нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения S опишите выигрышную стратегию Саши.

Постройте дерево всех партий, возможных при этой выигрышной стратегии Саши (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход, в узлах — количество камней в куче.

27. Для автоматизированной системы управления движением городских пассажирских автобусов требуется написать программу, которая решает следующую задачу.

На некоторой остановке в течение одного часа для каждого пассажирского автобуса фиксируется время прибытия в минутах (целое число от 0 до 60), номер маршрута (целое число), название предприятия (текстовая строка 20 символов). Все автобусы одного маршрута принадлежат одному предприятию; одно предприятие может обслуживать несколько маршрутов. Для каждого маршрута задан плановый интервал движения в минутах (целое число от 5 до 15) — промежуток времени между моментами прихода автобусов данного маршрута. Если автобусы некоторого маршрута допускают интервал движения, превышающий плановый более чем на 2 минуты, то на предприятие начисляется по одному штрафному баллу за каждую минуту.

Напечатать список маршрутов и предприятий, чьи автобусы допустили нарушения, и число штрафных баллов в виде <Номер маршрута> <Предприятие> <Число штрафных баллов>.

Исходные данные вводятся в компьютер в следующем порядке:

Сначала вводится число M — число маршрутов, проходящих через данную остановку, а затем вводится M строк вида <номер маршрута> <интервал движения> <название предприятия>.

Здесь <номер маршрута> — разные целые числа в количестве M , <интервал движения> — целые числа от 5 до 15, <название предприятия> — строка символов не более 20.

Далее вводится число N — число прошедших через остановку автобусов, затем вводится N строк вида <время прибытия> <номер маршрута>. <Время прибытия> — целые числа от 0 до 60, вводятся в порядке неубывания, <номер маршрута> — целые числа, каждое число обязательно совпадает с одним из <номеров маршрута>, введённых выше.

Вариант № 3

Часть 1

1. Для передачи чисел по каналу связи с помехами используется следующий алгоритм кодирования передаваемых чисел. Вместо каждой цифры записывается её двоичный 4-битовый код. Затем вычисляется остаток от деления суммы чисел исходного числа на 16. 4-битовое двоичное представление полученного числа дописывается в конец последовательности.

Например, при кодировании числа 965 получим код 1001011001010100. Здесь последние четыре цифры 0100 — двоичное представление числа 4 (остаток от деления суммы цифр числа $9 + 6 + 5 = 20$ на 16).

При передаче некоторого числа была получена последовательность 00101001011101010110. Известно, что в результате передачи произошла ошибка в одной позиции.

Определите, какое из чисел могло быть передано.

- 1) 3975 2) 2875 3) 2975 4) 2977

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	x_7	F
1	1	1	0	1	1	1	0
			1				1
	0		0				1

Каким из приведённых ниже выражений может быть F ?

- 1) $(x_1 \vee x_2 \vee x_3) \wedge x_4 \wedge (x_5 \vee x_6 \vee x_7)$
- 2) $\neg(x_1 \vee x_2 \vee x_3) \wedge x_4 \wedge \neg(x_5 \vee x_6 \vee x_7)$
- 3) $(x_1 \wedge x_2 \wedge x_3) \vee x_4 \vee (x_5 \wedge x_6 \wedge x_7)$
- 4) $\neg(x_1 \wedge x_2 \wedge x_3) \vee x_4 \vee \neg(x_5 \wedge x_6 \wedge x_7)$

3. Ниже представлены фрагменты двух таблиц из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведённых данных девичью фамилию бабушки Караваевой М.Н.

Таблица 1		
ID	Ф. И. О.	Пол
3	Караваев А.И.	м
12	Смирнова О.К.	ж
48	Петров С.Е.	м
5	Караваев Н.А.	м
14	Грицук А.Ю.	ж
2	Караваева М.Н.	ж
8	Круглов Ф.А.	м
9	Фёдорова Н.П.	ж
17	Круглова С.Ф.	ж
24	Грицук Ю.В.	м

Таблица 2	
ID_Родителя	ID_Ребёнка
3	48
12	48
3	5
12	5
5	2
14	2
8	17
9	17
11	45
24	14

- 1) Фёдорова Н.П.
3) Смирнова О.К.

- 2) Грицук Ю.В.
4) Круглова С.Ф.

4. Даны числа 65_{10} , 37_8 , 305_8 , $F3_{16}$. Сколько из этих чисел в двоичной записи будут содержать 5 и более единиц?

5. Между населёнными пунктами A , B , C , D , E и F построены дороги, протяжённость которых приведена в таблице. (Прочерк в таблице означает, что прямой дороги между соответствующими пунктами нет.)

	A	B	C	D	E	F
A	-	5	-	-	-	18
B	5	-	4	-	2	-
C	-	4	-	10	-	6
D	-	-	10	-	-	3
E	-	2	-	-	-	10
F	18	-	6	3	10	-

Определите длину кратчайшего пути между пунктами A и F (при условии, что передвигаться можно только по построенным дорогам).

6. У исполнителя *Калькулятор* имеется две команды:

- 1) **умножь на 3**
2) **прибавь 2**

Выполняя первую из них, *Калькулятор* умножает число на экране на 3, а выполняя вторую, прибавляет 2. Запишите порядок команд в программе получения из 2 числа 48, содержащей не более 5 команд, указывая

лишь номера команд. Например 12121 — это программа: **умножь** на 3, **прибавь** 2, **умножь** на 3, **прибавь** 2, **умножь** на 3, которая преобразует число 1 в число 51.

7. Дан фрагмент электронной таблицы в режиме отображения формул.

	А	В	С
1		4	7
2	$= (B1 + C1 + 4) * (C1 - 1)$	$= (B1 + 1) * C1 * B1 - 5$	$= A1 * 9$

Какое число должно быть записано в ячейке А1, чтобы построенная после выполнения вычислений диаграмма по значениям диапазона ячеек А2:С2 соответствовала рисунку 3?

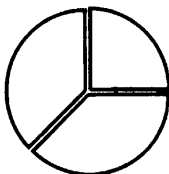


Рис. 3.

8. Определите, что будет напечатано в результате выполнения программы, записанной ниже на различных языках программирования.

Паскаль	Си
<pre>var s, k: integer; begin s := 1000; k := 5; repeat s := s - k; k := k + 5 until s < 230; writeln(s+k); end.</pre>	<pre>#include <stdio.h> void main() { int s, k; s = 1000; k = 5; do { s -= k; k += 5; } while (!(s < 230)); printf("%d", s+k); }</pre>
Бейсик	Алгоритмический язык
<pre>S = 1000 K = 5 DO S = S - K K = K + 5 LOOP UNTIL S<230 PRINT S+K</pre>	<pre>алг нач цел s, k s := 1000; k := 5 нц s := s - k; k := k + 5 кц при s < 230 вывод s+k кон</pre>

9. Документы передают с одного компьютера на другой следующим способом: сжать архиватором, передать архив по каналу связи, распаковать. Известны характеристики:

- средняя скорость передачи данных по каналу связи составляет 2^{23} бит в секунду;
- объём сжатого архиватором текстового документа равен 40% от исходного;
- объём сжатого архиватором графического документа равен 70% от исходного;
- время, требуемое на сжатие документа, — 10 секунд, а на распаковку — 1 секунда.

Передаются текстовый документ А объёмом 60 Мб и графический документ Б объёмом 40 Мб. В ответе напишите букву А, если документ А передадут быстрее, или Б, если быстрее передадут документ Б. Сразу после буквы напишите количество секунд, разница в доставке. Так, например, если документ Б дойдёт быстрее документа А на 23 секунды, в ответе нужно написать Б23.

10. Все четырёхбуквенные слова, составленные из букв А, В, Д, Л, записаны в алфавитном порядке. Вот начало списка:

1. АААА
2. АААВ
3. АААД
4. АААЛ
5. ААВА ...

Запишите слово, которое стоит на 197-м месте от начала списка.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Паскаль	Си
<pre> procedure F(n: integer); begin write(n, ' '); if n <> 1 then if n mod 2 = 0 then F(n+1); else F(n div 2) end </pre>	<pre> void F(int n) { printf("%d ", n); if (n != 1) if (n % 2 == 0) F(n+1); else F(n/2); } </pre>

Бейсик	Алгоритмический язык
<pre> SUB F(n) PRINT n, ' ' IF n <> 1 THEN IF n MOD 2 = 0 THEN F(n + 1) ELSE F(n \ 2) END IF END IF END SUB </pre>	<pre> алг F(цел n) нач вывод n, ' ' если n <> 1 то если mod(n,2)=0 то F(n+1) иначе F(div(n,2)) все все кон </pre>

Чему равна сумма чисел, напечатанных на экране при выполнении вызова F(13)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 218.128.144.7

Маска: 255.255.224.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
0	7	255	224	218	144	128	116

Пример. Пусть искомый IP-адрес 192.168.128.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Необходимый для регистрации на сайте логин каждого пользователя состоит из 7 символов. Первые пять символов пользователь придумывает сам из 17 фиксированных букв латинского алфавита. Следующие два

символа — десятичные цифры в любом порядке — автоматически дписываются программой.

Каждый такой код в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит).

Определите объём памяти (в байтах), отводимый этой программой для записи 50 пользователей. В ответе запишите только число, слово «байт» писать не нужно.

14. Исполнитель *Чертёжник* перемещается на координатной плоскости, оставляя след в виде линии. *Чертёжник* может выполнять команду **Сместиться на (a, b)** (где a, b — целые числа), перемещающую *Чертёжника* из точки с координатами (x, y) в точку с координатами $(x+a, y+b)$. Если числа a, b положительные, то значение соответствующей координаты увеличивается, если отрицательные — уменьшается. Например, если *Чертёжник* находится в точке с координатами $(4, 2)$, то команда **Сместиться на $(2, -3)$** переместит *Чертёжника* в точку $(6, -1)$.

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что последовательность команд будет выполнена указанное число раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами n, a, b обозначены неизвестные числа, при этом $n > 1$):

НАЧАЛО

Сместиться на $(-14, 16)$

ПОВТОРИ n РАЗ

Сместиться на $(22, -15)$

Сместиться на (a, b)

Сместиться на $(-9, 10)$

КОНЕЦ ПОВТОРИ

Сместиться на $(-7, -30)$

КОНЕЦ

Укажите наименьшее возможное значение числа n , для которого найдутся такие значения чисел a и b , что после выполнения программы *Чертёжник* возвратится в исходную точку, из которой он начал движение.

15. На рисунке 4 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?

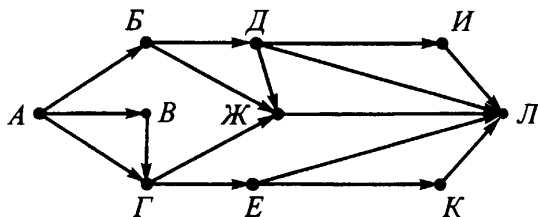


Рис. 4.

16. Сколько единиц содержится в двоичной записи значения выражения: $16^{500} + 4^{600} - 16^?$

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Акварель	310
Натюрморт	360
Урок	270
Урок Акварель	470
Натюрморт & (Урок Акварель)	180
Акварель Натюрморт	530

Сколько страниц (в сотнях тысяч) будет найдено по запросу
Урок & (Акварель | Натюрморт)?

18. На числовой прямой даны два отрезка: $P = [2, 28]$ и $Q = [11, 36]$. Укажите наименьшую возможную длину такого отрезка A , что логическое выражение

$$(\neg(x \in A) \vee \neg((x \in Q) \wedge (x \in P))) \rightarrow \neg(x \in A)$$

тождественно истинно, то есть принимает значение 1 при любом значении переменной x .

19. В программе обрабатывается одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8; 5; -11; 4; -6; -9; 4; -10; 7; 8 соответственно, т.е. $A[0] = 8$; $A[1] = 5$ и т.д. Определите значение переменной k после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> k = 0 FOR i = 0 TO 8 IF A(i)*A(i+1)>0 THEN k = k + 1 : t = A(i+1) A(i+1) = A(9-k+1) A(9-k+1) = t END IF NEXT i </pre>	<pre> k:=0 нц для i от 0 до 8 если A[i]*A[i+1]>0 то k:=k+1; t:= A[i+1] A[i+1]:=A[9-k+1] A[9-k+1]:=t все кц </pre>
Паскаль	Си
<pre> k:=0; for i:=0 to 8 do if A[i]*A[i+1]>0 then begin k:=k+1; t:= A[i+1]; A[i+1]:=A[9-k+1]; A[9-k+1]:=t end end </pre>	<pre> k=0; for (i=0; i<9; i++) do if (A[i]*A[i+1]>0) { k++; t=A[i+1]; A[i+1]=A[9-k+1]; A[9-k+1]=t; } } </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа K и R . Укажите наибольшее из таких чисел x , при вводе которых алгоритм печатает сначала 4, а потом 5.

Бейсик	Паскаль
<pre> INPUT X K = 0 : R = 9 Y = X MOD 10 WHILE X > 0 K = K + 1 IF R > X MOD 10 THEN R = X MOD 10 END IF X = X \ 10 WEND R = Y - R PRINT K PRINT R </pre>	<pre> var x, K, R, y: integer; begin readln(x); K := 0; R := 9; y := x mod 10; while x > 0 do begin K := K + 1; if R > x mod 10 then R := x mod 10; x := x div 10 end; R := y - R; writeln(K); writeln(R) end. </pre>

Алгоритмический язык	Си
алг нач цел x, K, R, y ввод x $K := 0$ $R := 9$ $y := \text{mod}(x, 10)$ нц пока $x > 0$ $K := K + 1$ если $R > \text{mod}(x, 10)$ то $R := \text{mod}(x, 10)$ все $x := \text{div}(x, 10)$ кц $R := y - R$ вывод $K, \text{нс}, R$ кон	<pre>#include <stdio.h> void main() { int x, K, R, y; scanf ("%d", &x); K = 0; R = 9; y = x % 10; while (x > 0) { K = K + 1; if (R > x % 10) R = x % 10; x = x / 10; } R = y - R; printf("%d\n%d", K, R); }</pre>

21. Ниже на четырёх языках записан алгоритм. Получив на вход целое число k , этот алгоритм печатает число i . Укажите число различных значений входной переменной k , при которых программа выдаёт тот же ответ, что и при входном значении $k = 100$. Значение $k = 100$ также включается в подсчёт различных значений k .

Бейсик	Алгоритмический язык
INPUT K $I = 5$ WHILE $I < 100$ AND $F(K) \geq I$ $I = I + 1$ WEND PRINT I FUNCTION $F(X)$ $F = \text{SQR}(X)$ END FUNCTION	алг нач цел i, k ввод k $i := 5$ нц пока $i < 100$ и $f(k) \geq i$ $i := i + 1$ кц вывод i кон алг вещь $F(\text{цел } x)$ нач знач $:= \text{sqrt}(x)$ кон

Паскаль	Си
<pre> var k,i: integer; Function F(x:integer):real; begin f:=sqrt(x) end; begin readln(k); i:=5; while (i<100) and (f(k)>=i) do i:=i+1; writeln(i) end.</pre>	<pre> #include<stdio.h> #include<math.h> float F(int x) { return sqrt(x); } void main() { int k, i; scanf("%d", &k); i=5; while (i<100 && F(k)>=i) i++; printf("%d", i); }</pre>

22. У исполнителя P148 три команды, которым присвоены номера:

1. прибавь 1,
2. умножь на 4,
3. умножь на 8.

Первая из них увеличивает число на экране на 1, вторая — в 4 раза, а третья — в 8 раз. Программа для исполнителя P148 — это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 28?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$(\neg x_1 \vee \neg x_2) \wedge ((x_1 \wedge x_2) \rightarrow x_3) \wedge (\neg x_3 \equiv y_1) = 1,$$

$$(\neg x_2 \vee \neg x_3) \wedge ((x_2 \wedge x_3) \rightarrow x_4) \wedge (\neg x_4 \equiv y_2) = 1,$$

...

$$(\neg x_6 \vee \neg x_7) \wedge ((x_6 \wedge x_7) \rightarrow x_8) \wedge (\neg x_8 \equiv y_6) = 1,$$

$$(\neg x_7 \vee \neg x_8) \wedge (\neg y_7 \vee y_8) = 1.$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается два натуральных числа N и M одинаковой размерности, не превосходящие 10^9 . Программа должна найти наибольшую цифру в каждом из чисел и вывести на экран сумму найденных цифр. (Например, для чисел $N = 216$, $M = 471$ программа должна вывести 13. Программист торопился и написал программу неправильно.

Алгоритмический язык

```

алг
нач
    цел N, M, MaxN, MaxM
    ввод N
    ввод M
    MaxN:=0; MaxM:=0
    нц пока N > 10 или M > 10
        N:=div(N,10)
        M:=div(N,10)
        если MaxN < mod(N,10) то
            MaxN:=mod(N,10)   все
        если MaxM < mod(M,10) то
            MaxM:=mod(N,10)   все
    кц
    вывод MaxN+MaxM
кон

```

Си

```

#include <stdio.h>
void main() {
    long N,M;
    int MaxN, MaxM;
    scanf("%ld %ld", &N, &M);
    MaxN = 0; MaxM = 0;
    while (N>10 || M>10) {
        N /= 10;
        M /= 10;
        if MaxN > N % 10
            MaxN = N % 10;
        if MaxM > M % 10
            MaxM = N % 10;
    }
    printf("%d", MaxN+MaxM);
}

```

Паскаль

```
var N,M: longint;  
    MaxN, MaxM: integer;  
begin  
    readln(N, M);  
    MaxN:=0;  
    MaxM:=0;  
    while (N > 10) or (M > 10) do begin  
        N:= N div 10;  
        M:= M div 10;  
        if MaxN < N mod 10 then  
            MaxN:=N mod 10;  
        if MaxM < M mod 10 then  
            MaxM:=N mod 10;  
    end;  
    writeln(MaxN+MaxM)  
end.
```

Бейсик

```
DIM N AS LONG  
DIM M AS LONG  
INPUT N  
INPUT M  
MaxN = 0; MaxM = 0;  
WHILE N > 10 OR M > 10  
    N = N \ 10  
    M = M \ 10  
    IF MaxN < N MOD 10 THEN  
        MaxN = N MOD 10  
    END IF  
    IF MaxM < M MOD 10 THEN  
        MaxM = N MOD 10  
    - END IF  
WEND  
PRINT MaxN+MaxM  
END
```

Последовательно выполните следующее:

1. Напишите, что выведет программа при вводе чисел 5426 и 8317.
2. Приведите пример двух таких чисел, что, несмотря на ошибки, программа печатает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, — приведите правильный вариант строки.
4. Найдите и выпишите строку, которую можно убрать для улучшения эффективности работы программы (их может быть одна или несколько). Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан массив из 40 элементов. Каждый элемент может принимать целочисленные значения от 0 до 300. Опишите на русском языке или на одном из языков программирования алгоритм, который позволит найти произведение тех элементов массива, которые имеют чётный индекс и кратны семи. Гарантируется, что хотя бы один элемент массива удовлетворяет этому условию. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Бейсик
<pre>const N=40; var a: array [0..N-1] of integer; i, p: integer; begin for i:=0 to N-1 do readln(a[i]); ... end.</pre>	<pre>CONST N = 40 DIM A(N) AS INTEGER DIM I AS INTEGER DIM P AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>

Си	Естественный язык
<pre>#include <stdio.h> #define N 40 void main(void){ int a[N], i, p; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 40 элементов. Объявляем целочисленные переменные I, P. В цикле от 1 до 40 вводим элементы массива А с 1-го по 40-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого вами языка программирования.

26. Два игрока, Коля и Саша, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Коля. За один ход игрок может добавить в кучу один камень или увеличить количество камней в куче в три раза. Например, имея кучу из 8 камней, за один ход можно получить кучу из 9 или 24 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней. Игра завершается в тот момент, когда количество камней в куче становится не менее 109. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 109 или больше камней. В начальный момент в куче было S камней, $1 \leq S \leq 108$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обоснуйте свой ответ.

1. а) Укажите все такие значения числа S , при которых Коля может выиграть в один ход. Обоснуйте, что найдены все нужные значения S , и укажите выигрышный ход для каждого указанного значения S .

б) Укажите такое значение S , при котором Коля не может выиграть за один ход, но при любом ходе Коли Саша может выиграть своим первым ходом. Опишите выигрышную стратегию Саши.

2. Укажите два таких значения S , при которых у Коли есть выигрышная стратегия, причём

— Коля не может выиграть за один ход, и

— Коля может выиграть своим вторым ходом независимо от того, как будет ходить Саша. Для каждого указанного значения S опишите выигрышную стратегию Коли.

3. Укажите значение S , при котором

— у Саши есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Коли, и

— у Саши нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения S опишите выигрышную стратегию Саши. Постройте дерево всех партий, возможных при этой выигрышной стратегии Саши (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход, в узлах — количество камней в куче.

27. Завод по огранке драгоценных камней приобрёл сейф повышенной надёжности. Для определения драгоценных камней, которые необходимо положить в сейф, сначала отбираются 5% самых дорогих камней.

Если у самого дешёвого камня из вошедших в группу 5% самых дорогих ценовая категория оказывается такой же, как и у нескольких других, то эти камни тоже включаются в группу камней для размещения в сейфе повышенной надёжности в том случае, если их ценовая категория не менее 15.

Напишите эффективную по времени работы и по используемой памяти программу, которая по результатам входных данных будет определять, какую минимальную цену должен иметь драгоценный камень, чтобы его поместили в сейф повышенной надёжности.

На вход программе сначала подаётся общее количество камней на складе N . В каждой из следующих N строк находится информация по каждому камню отдельно в следующем формате:

<Название драгоценного камня> <Код> <Ценовая категория>,

где

<Название драгоценного камня> — строка, состоящая не более чем из 20 символов,

<Код> — строка, состоящая не более чем из 15 символов,

<Ценовая категория> — целое число от 1 до 20.

<Название драгоценного камня>, <Код> и <Ценовая категория> разделены одним пробелом.

Пример входной строки:

Рубин Р1234 13

Программа должна выводить минимальную ценовую категорию драгоценного камня, который необходимо положить в сейф повышенной надёжности. (Гарантируется, что хотя бы один камень удовлетворяет условиям и будет положен в сейф.)

Вариант № 4

Часть 1

1. Для передачи чисел по каналу связи с помехами используется следующий алгоритм кодирования передаваемых чисел. Вместо каждой цифры записывается её двоичный 4-битовый код. Затем вычисляется остаток от деления суммы чисел исходного числа на 16. 4-битовое двоичное представление полученного числа дописывается в конец последовательности.

Например, при кодировании числа 965 получим код 1001011001010100. Здесь последние четыре цифры 0100 — двоичное представление числа 4 (остаток от деления суммы цифр числа $9 + 6 + 5 = 20$ на 16). При передаче некоторого числа была получена последовательность

01000101011110000111.

Известно, что в результате передачи произошла ошибка в одной позиции. Определите, какое из чисел могло быть передано.

1) 4578

2) 4178

3) 4568

4) 6578

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	x_7	F
1			0	1			1
	0		0		0		1
0	1	0	1				0

Каким из приведённых ниже выражений может быть F ?

- 1) $(x1 \vee \neg x2 \vee x3) \wedge \neg x4 \wedge (x5 \vee \neg x6 \vee x7)$
- 2) $(\neg x1 \vee x2 \vee \neg x3) \wedge x4 \wedge (\neg x5 \vee x6 \vee \neg x7)$
- 3) $(x1 \wedge \neg x2 \wedge x3) \vee x4 \vee (x5 \wedge \neg x6 \wedge x7)$
- 4) $(\neg x1 \wedge x2 \wedge \neg x3) \vee \neg x4 \vee (\neg x5 \wedge x6 \wedge \neg x7)$

3. Ниже представлены фрагменты двух таблиц из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведённых данных фамилию и инициалы тётки Романовой У.Т.

Пояснение: тёткой считается родная сестра отца или матери.

Таблица 1		
ID	Ф. И. О.	Пол
64	Терещенко И.М.	ж
33	Егоров Г.В.	м
34	Романова У.Т.	ж
35	Зими́на В.В.	ж
23	Романов Т.А.	м
27	Егорова А.Г.	ж
48	Маховец Д.Г.	м
53	Егорова Л.С.	ж
67	Маховец В.Д.	ж
58	Сотник А.Г.	ж
59	Абрамян И.К.	м
76	Хоненко Н.В.	ж
71	Рубан Г.В.	м

Таблица 2	
ID_Родителя	ID_Ребёнка
23	34
27	34
34	35
76	71
64	23
71	23
64	67
59	12
11	45
33	27
53	64
34	76
59	48

- 1) Сотник А.Г.
- 2) Терещенко И.М.
- 3) Егорова А.Г.
- 4) Маховец В.Д.

4. Даны числа 32_{10} , 51_8 , $A1_{16}$, $8F_{16}$. Сколько из этих чисел в двоичной записи будут содержать 5 и более единиц?

5. Между населёнными пунктами A, B, C, D, E и F построены дороги, протяжённость которых приведена в таблице. (Прочерк в таблице означает, что прямой дороги между соответствующими пунктами нет.)

	A	B	C	D	E	F
A	-	8	-	-	-	-
B	8	-	3	6	2	-
C	-	3	-	-	9	5
D	-	6	-	-	4	3
E	-	2	9	4	-	12
F	-	-	5	3	12	-

Определите длину кратчайшего пути между пунктами A и F (при условии, что передвигаться можно только по построенным дорогам).

6. У исполнителя *Калькулятор* имеется две команды:

- 1) возведи в квадрат;
- 2) прибавь 3.

Выполняя первую из них, *Калькулятор* возводит число на экране в квадрат, а выполняя вторую, увеличивает его на 3. Запишите порядок команд в программе получения из 3 числа 225, содержащей не более 4 команд, указывая лишь номера команд. Например 1122 — это программа: возведи в квадрат, возведи в квадрат, прибавь 3, прибавь 3, которая преобразует число 2 в число 22.

7. Дан фрагмент электронной таблицы в режиме отображения формул.

	A	B	C
1	6		5
2	$= (C1 - 3) * 2$	$= 4 * (A1 - 2 * B1)$	$= A1 * B1$

Какое целое число должно быть записано в ячейке B1, чтобы построенная после выполнения вычислений диаграмма по значениям диапазона ячеек A2:C2 соответствовала рисунку 5?

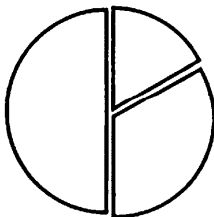


Рис. 5.

8. Определите, что будет напечатано в результате выполнения программы, записанной ниже на различных языках программирования.

Бейсик	Алгоритмический язык
<pre> S = 10 K = 50 DO S = S + K K = K - 15 LOOP UNTIL K<10 PRINT S+K </pre>	<pre> алг нач цел s, k s := 10; k := 50 нц s := s + k; k := k - 15 кц при k < 10 вывод s+k кон </pre>
Паскаль	Си
<pre> var s, k: integer; begin s := 10; k := 50; repeat s := s + k; k := k - 15 until k < 10; writeln(s+k); end. </pre>	<pre> #include <stdio.h> void main() { int s, k; s = 10; k = 50; do { s += k; k -= 15; } while (!(k < 10)); printf("%d", s+k); } </pre>

9. Документы передают с одного компьютера на другой следующим способом: сжимают документ архиватором, создавая архив, передают архив по каналу связи и затем распаковывают. Известны следующие характеристики передачи файлов:

— средняя скорость передачи данных по каналу связи составляет 2^{22} бит в секунду;

— объём сжатого архиватором текстового документа равен 45% от исходного;

— объём сжатого архиватором графического документа равен 60% от исходного;

— время, требуемое на сжатие документа, — 10 секунд, а на распаковку — 1 секунда.

Передаются текстовый документ А объёмом 70 Мб и графический документ Б объёмом 90 Мб. В ответе напишите букву А, если документ А передадут быстрее, или Б, если быстрее передадут документ Б. Сразу после буквы напишите количество секунд — разницу во времени передачи документов. Так, например, если документ Б дойдёт быстрее документа А на 23 секунды, в ответе нужно написать Б23.

10. Все четырёхбуквенные слова, составленные из букв А, К, М, Я, записаны в алфавитном порядке. Вот начало списка:

1. АААА
2. АААК
3. АААМ
4. АААЯ
5. ААКА...

Запишите слово, которое стоит на 142-м месте от начала списка.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> SUB F(n) PRINT n, ' ' IF n <> 1 THEN IF n MOD 3 = 0 THEN F(n \ 3) ELSE F(n - 1) END IF END IF END SUB </pre>	<pre> алг F(цел n) нач вывод n, ' ' если n <> 1 то если mod(n,3)=0 то F(div(n,3)) иначе F(n - 1) все все кон </pre>
Паскаль	Си
<pre> procedure F(n: integer); begin write(n, ' '); if n <> 1 then if n mod 3 = 0 then F(n div 3) else F(n - 1) end; end; </pre>	<pre> void F(int n) { printf("%d ", n); if (n != 1) if (n % 3 == 0) F(n/3); else F(n-1); } </pre>

Чему равна сумма чисел, напечатанных на экране при выполнении вызова F(16)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 219.22.108.16

Маска: 255.255.224.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
255	96	16	216	22	219	224	0

Пример. Пусть искомый IP-адрес 168.53. 224.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	15	53	0	17	224

В этом случае правильный ответ будет записан в виде: BEHF.

13. Для регистрации на сайте необходимо ввести следующую информацию: логин, состоящий из букв английского алфавита длиной 15 символов (всего используется 20 различных символов); пароль, состоящий из цифр (от 0 до 9) и длиной 10 символов; пол (М или Ж). Каждая такая учётная запись кодируется минимально возможным и одинаковым (целым) количеством байт (при этом используют посимвольное кодирование, и все символы логина кодируются одинаковым и минимально возможным количеством бит, все символы пароля кодируются одинаковым и минимально возможным количеством, и пол также кодируется минимально возможным количеством бит). Определите объём памяти (в байтах), необходимый для хранения 20 записей. В ответе запишите только число, слово «байт» писать не нужно.

14. Исполнитель *Чертёжник* перемещается на координатной плоскости, оставляя след в виде линии. *Чертёжник* может выполнять команду *Сместиться на (a,b)* (где a, b — целые числа), перемещающую *Чертёжника* из точки с координатами (x,y) в точку с координатами $(x+a, y+b)$. Если числа a, b положительные, то значение соответствующей

координаты увеличивается, если отрицательные — уменьшается. Например, если *Чертёжник* находится в точке с координатами (4, 2), то команда *Сместиться* на (2, -3) переместит *Чертёжника* в точку (6, -1).

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что *последовательность команд* будет выполнена указанное *число* раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами n , a , b обозначены неизвестные числа, при этом $n > 1$):

НАЧАЛО

Сместиться на (-28, -35)

ПОВТОРИ n РАЗ

Сместиться на (a , b)

Сместиться на (-11, 16)

Сместиться на (32, 24)

КОНЕЦ ПОВТОРИ

Сместиться на (10, 11)

КОНЕЦ

Укажите наименьшее возможное значение числа n , для которого найдутся такие значения чисел a и b , что после выполнения программы *Чертёжник* возвратится в исходную точку, из которой он начал движение.

15. На рисунке 6 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?

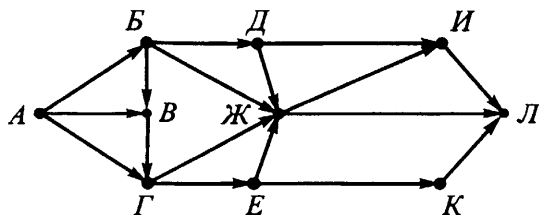


Рис. 6.

16. Сколько единиц содержится в двоичной записи значения выражения:
 $8^{800} + 2^{1020} - 32$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Недоросль	810
Фонвизин	620
герои	380
Фонвизин герои	1000
Недоросль & герои	120
(Фонвизин герои) & Недоросль	280

Сколько страниц (в сотнях тысяч) будет найдено по запросу
Недоросль & Фонвизин?

18. На числовой прямой даны два отрезка: $P = [6, 35]$ и $Q = [24, 46]$. Укажите наименьшую возможную длину такого отрезка A , что логическое выражение

$$((x \in A) \vee ((x \in Q) \wedge (x \in P))) \rightarrow (x \in A)$$

тождественно истинно, то есть принимает значение 1 при любом значении переменной x .

19. В программе обрабатывается одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8; 5; -11; 4; -6; -9; 4; -10; 7; 8 соответственно, т.е. $A[0] = 8$; $A[1] = 5$ и т.д.

Определите значение переменной k после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> k = 0 FOR i = 0 TO 8 IF A(i)*A(i+1)<0 THEN k = k + 1 : t = A(i) A(i) = A(9-i) A(9-i) = t END IF NEXT i </pre>	<pre> k:=0 нц для i от 0 до 8 если A[i]*A[i+1]<0 то k:=k+1 : t:= A[i] A[i]:=A[9-i] A[9-i]:=t все кц </pre>

Паскаль	Си
<pre> k:=0; for i:=0 to 8 do if A[i]*A[i+1]<0 then begin k:=k+1; t:= A[i]; A[i]:=A[9-i]; A[9-i]:=t end </pre>	<pre> k=0; for (i=0; i<9; i++) do if (A[i]*A[i+1]<0) { k++; t=A[i]; A[i]=A[9-i]; A[9-i]=t; } </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа K и R . Укажите наименьшее из таких чисел x , при вводе которых алгоритм печатает сначала 4, а потом 6.

Алгоритмический язык	Си
<pre> алг нач цел x, K, R, y ввод x K := 0 R := 0 нц пока x > 0 K := K + 1 если R < mod(x, 10) то R := mod(x, 10) все если x<10 то y:=x все x := div(x, 10) кц R := R - y вывод K, nc, R кон </pre>	<pre> #include <stdio.h> void main() { int x, K, R, y; scanf ("%d", &x); K = 0; R = 0; while (x > 0) { K = K + 1; if (R < x % 10) R = x % 10; if (x < 10) y = x; x = x / 10; } R = R - y; printf("%d\n%d", K, R); } </pre>

Бейсик	Паскаль
<pre> INPUT X K = 0 : R = 0 WHILE X > 0 K = K + 1 IF R < X MOD 10 THEN R = X MOD 10 END IF IF X < 10 THEN Y = X X = X \ 10 WEND R = R - Y PRINT K PRINT R </pre>	<pre> var x, K, R, y: integer; begin readln(x); K := 0; R := 0; while x > 0 do begin K := K + 1; if R < x mod 10 then R := x mod 10; if x < 10 then y := x; x := x div 10 end; R := R - y; writeln(K); writeln(R) end. </pre>

21. Ниже на четырёх языках записан алгоритм. Получив на вход целое число k , этот алгоритм печатает число t . Укажите число различных значений входной переменной k , при которых программа выдаёт тот же ответ, что и при входном значении $k = 10$. Значение $k = 10$ также включается в подсчёт различных значений k .

Бейсик	Алгоритмический язык
<pre> INPUT K T = 0 WHILE T < 10 AND F(K) >= T T = T + 0.1 WEND PRINT I FUNCTION F (X) F = 1/X END FUNCTION </pre>	<pre> алг нач цел k; вещ t ввод k t:=0 нц пока i<1 и f(k)>=t t:=t+0.1 кц вывод t кон алг вещ F(цел x) нач знач := 1/x кон </pre>

Паскаль	Си
<pre>var k: integer; t:real; Function F(x:integer):real; begin f:=sqrt(x) end; begin readln(k); t:=0; while (t<1) and (f(k)>=i) do t:=t+0.1; writeln(t:2:1) end.</pre>	<pre>#include<stdio.h> #include<math.h> float F(int x) { return (float)1/x; } void main() { int k; float t=0; scanf("%d", &k); while (t<1 && F(k)>=t) t+=0.1; printf("%f", t); }</pre>

22. У исполнителя P139 три команды, которым присвоены номера:

1. прибавь 1,
2. умножь на 3,
3. умножь на 9.

Первая из них увеличивает число на экране на 1, вторая — в 3 раза, а третья — в 9 раз. Программа для исполнителя P139 — это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 28?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$(\neg x_1 \rightarrow x_2) \wedge \neg(x_1 \rightarrow \neg x_3) \wedge (\neg x_2 \rightarrow y_1) = 1,$$

$$(\neg x_2 \rightarrow x_3) \wedge \neg(x_2 \rightarrow \neg x_4) \wedge (\neg x_3 \rightarrow y_2) = 1,$$

...

$$(\neg x_6 \rightarrow x_7) \wedge \neg(x_6 \rightarrow \neg x_8) \wedge (\neg x_7 \rightarrow y_6) = 1,$$

$$(\neg x_7 \rightarrow x_8) \wedge \neg(x_7 \rightarrow \neg y_7) = 1,$$

$$(\neg y_7 \rightarrow y_8) = 1.$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается два натуральных числа N и M одинаковой размерности, не превосходящие 10^9 .

Программа должна вычислить сумму старших разрядов этих чисел, сумму их младших разрядов и вывести наибольшую из найденных сумм. Если суммы одинаковы, то одну из них. (Например, для чисел $N = 215$, $M = 416$ программа должна вывести 11.)

Программист торопился и написал программу неправильно.

Алгоритмический язык	
алг	
нач	
цел N, M, S	
ввод N	
ввод M	
$S := \text{mod}(N+M, 10)$	
нц пока $N > 0$	
$N := \text{div}(N, 10)$	
нц пока $M > 10$	
$M := \text{div}(N, 10)$	
кц	
кц	
если $S > N+M$ то	
вывод S	
иначе	
вывод $N+M$	
все	
кон	

Паскаль

```
var N,M: longint;  
    S: integer;  
begin  
    readln(N, M);  
    S:= (N + M) mod 10;  
    while N > 0 do  
    begin  
        N:= N div 10;  
        while M > 10 do  
            M:= M div 10  
        end;  
        if S > N + M then  
            writeln(S)  
        else  
            writeln(M+N)  
        end;  
    end.  
end.
```

Бейсик

```
DIM N AS LONG  
DIM M AS LONG  
INPUT N  
INPUT M  
S = (N + M) MOD 10  
WHILE N > 0  
    N = N \ 10  
    WHILE M > 10  
        M = M \ 10  
    WEND  
WEND  
IF S > N + M THEN  
    PRINT S  
ELSE  
    PRINT N+M  
END IF  
END
```

Си

```
#include <stdio.h>
void main() {
    long N,M;
    int S;
    scanf("%ld %ld", &N, &M);
    S = (N + M) % 10;
    while (N>0) {
        N /= 10;
        while (M>10)
            M /= 10;
    }
    if S > N + M
        printf("%d", S);
    else
        printf("%d", N+M);
}
```

Последовательно выполните следующее:

1. Напишите, что выведет программа при вводе чисел 517 и 834.
2. Приведите пример двух таких чисел, что, несмотря на ошибки, программа печатает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, — приведите правильный вариант строки.
4. Найдите и выпишите строку, которую можно убрать для улучшения эффективности работы программы (их может быть одна или несколько). Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан массив из 30 элементов. Каждый элемент может принимать целочисленные значения от -100 до 100 . Опишите на русском языке или на одном из языков программирования алгоритм, который позволит подсчитать среднее арифметическое положительных элементов массива, которые кратны первому элементу массива (то есть делятся на него без остатка). Гарантируется, что первый элемент отличен от нуля. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Бейсик
<pre>const N=30; var a: array [0..N-1] of integer; i,k: integer; sum: real; begin for i:=0 to N-1 do readln(a[i]); ... end.</pre>	<pre>CONST N = 30 DIM A(N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM SUM AS SINGLE FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>
Си	Естественный язык
<pre>#include <stdio.h> #define N 30 void main(void){ int a[N]; int i, k; float sum; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 30 элементов. Объявляем целочисленные переменные I, K и вещественную переменную SUM. В цикле от 1 до 30 вводим элементы массива А с 1-го по 30-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования) или в виде блок-схемы. В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого вами языка программирования.

26. Два игрока, Коля и Саша, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Коля. За один ход игрок может добавить в кучу два камня или увеличить количество камней в куче в два раза. Например, имея кучу из 8 камней, за один ход можно получить кучу из 10 или 16 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней. Игра завершается в тот момент, когда количество камней в куче становится не менее 32. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 32 или больше камней. В начальный момент в куче было S камней, $1 \leq S \leq 28$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обоснуйте свой ответ.

1. а) Укажите все такие значения числа S , при которых Коля может выиграть в один ход. Обоснуйте, что найдены все нужные значения S , и укажите выигрышный ход для каждого указанного значения S .

б) Укажите такое значение S , при котором Коля не может выиграть за один ход, но при любом ходе Коли Саша может выиграть своим первым ходом. Опишите выигрышную стратегию Саши.

2. Укажите два таких значения S , при которых у Коли есть выигрышная стратегия, причём

- Коля не может выиграть за один ход, и
- Коля может выиграть своим вторым ходом независимо от того, как будет ходить Саша. Для каждого указанного значения S опишите выигрышную стратегию Коли.

3. Укажите значение S , при котором

- у Саши есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Коли, и
- у Саши нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения S опишите выигрышную стратегию Саши. Постройте дерево всех партий, возможных при этой выигрышной стратегии Саши (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход, в узлах — количество камней в куче.

27. По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число R , удовлетворяющее следующим условиям:

1) R — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел; произведения различных элементов последовательности, равных по величине, допускаются);

2) R делится на 14.

Если такого числа R нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены. Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

Вычисленное контрольное значение: ...

Контроль пройден (или Контроль не пройден)

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N . В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

Пример входных данных:

```
4
70
25
32
12
2240
```

Пример выходных данных для приведённого выше примера входных данных:

Вычисленное контрольное значение: 2240

Контроль пройден

Вариант № 5

Часть 1

1. Для кодирования некоторой последовательности, состоящей из букв русского алфавита, применили алгоритм шифра Вижинера:

1) Символы исходного алфавита нумеруются, начиная с нуля:

А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
20	21	22	23	24	25	26	27	28	29	30	31	32

2) Задаётся ключ кодирования — слово в исходном алфавите, например, МЕЛ.

3) Выписываются числа, соответствующие буквам кодируемого сообщения. Например, для сообщения КОМПЬЮТЕР получим последовательность чисел:

11 15 13 16 29 31 19 5 17

4) Под каждым числом полученной последовательности выписываются числа символов ключевого слова (в нашем случае, МЕЛ — 13 5 12), повторяющиеся до тех пор, пока не закончатся числа кодируемого слова :

11 15 13 16 29 31 19 5 17

13 5 12 13 5 12 13 5 12

5) В каждом столбце находят остаток от деления суммы его чисел на 33 — число символов исходного алфавита (для русского алфавита — 33).

Суммы чисел 24 20 25 29 34 43 32 10 29

Остатки от деления сумм на 33 24 20 25 29 1 10 32 10 29

6) Полученный числовой ряд преобразуется в символы исходного алфавита:

Ч У Ш Ъ Б Й Я Ъ

В результате кодирования некоторого сообщения с помощью данного алгоритма была получена последовательность символов ПОЯТЛО. В качестве ключа было использовано слово ЗОНТ. Укажите закодированное сообщение.

1) ПРИВЕТ 2) ЗАСАДА 3) ДОРОГА 4) ГЛОБУС

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	F
1	0	1	1	0	1	0
1	1	1	0	0	1	1
1	1	1	0	0	0	0

Каким выражением может быть F ?

- 1) $\neg x_1 \vee (x_2 \wedge x_3 \wedge x_4 \wedge x_5) \vee \neg x_6$
- 2) $\neg x_1 \wedge (x_2 \vee x_3 \vee x_4 \vee x_5) \wedge \neg x_6$
- 3) $x_1 \vee (x_2 \wedge \neg x_3 \wedge \neg x_4 \wedge x_5) \vee x_6$
- 4) $x_1 \wedge (x_2 \vee \neg x_3 \vee \neg x_4 \vee x_5) \wedge x_6$

3. Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы: символ «?» (вопросительный знак) означает ровно один произвольный символ; символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

В каталоге находятся шесть файлов:

even.chm
 devenv.ctm
 service.asmx
 esvect.tpm
 verve.thmx
 services.htm

Определите, по какой из масок из них будет отображена указанная группа файлов:

devenv.ctm
 service.asmx
 verve.thmx
 services.htm

- 1) $*e?v*.m*$
- 2) $?ev*.m?$
- 3) $?e?v*.m*$
- 4) $*ev?.?m*$

4. Как представляется число 111_{10} в двоичной системе счисления?

5. На рисунке 7 изображена схема дорог между населёнными пунктами и обозначена стоимость перевозки пассажиров между соседними населёнными пунктами. Определите наименьшую стоимость перевозки из пункта A в пункт B .

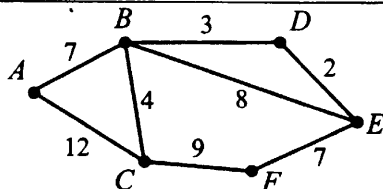


Рис. 7.

6. Дан алгоритм преобразования одного двузначного числа в другое.

1. Найти сумму цифр заданного числа и приписать её в конец числа.
2. Если полученное число чётно, прибавить к нему 10, если нечётно — прибавить 1.
3. В полученном числе оставить последние две цифры, остальные отбросить.

Укажите число, из которого в результате выполнения алгоритма будет получено число 56.

7. Дан фрагмент электронной таблицы.

	A	B	C	D
1	5	1	=МИН(A1:A4)	3
2	8	2		6
3	3	3		9
4	9	4		12

Значения ячеек D1:D4 были получены следующим образом: в ячейку D1 записали формулу, затем скопировали её в ячейки диапазона D2:D4.

Определите, какая формула была записана в ячейку D1.

- 1) = B1 * \$C1
- 2) = B1 * C\$1
- 3) = \$B1 * \$C1
- 4) = B\$1 * C\$1

8. Определите значение переменной k после выполнения фрагмента программы, записанной ниже на различных языках программирования.

Паскаль	Си
<pre>s := 0; k := 1; while s < 200 do begin s := s + 3 * k; if s < 200 then k := k + 1; end; writeln(k);</pre>	<pre>s = 0; k = 1; while (s < 200) { s += 3 * k; if (s < 200) k += 1; } printf("%d", k);</pre>

Бейсик	Алгоритмический язык
<pre> S = 0 K = 1 WHILE S < 200 S = S + 3 * K IF S < 200 THEN K = K + 1 END IF WEND PRINT K </pre>	<pre> s := 0 k := 1 нц пока s < 200 s := s + 3 * k если s < 200 то k := k + 1 все кц вывод k </pre>

9. Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 64 кГц и 16-битным разрешением. Запись длится 3 минуты, её результаты записываются в файл, сжатие данных не производится. Определите приблизительно размер полученного файла (в Мбайтах).

В качестве ответа укажите ближайшее к размеру полученного файла целое число, кратное 10.

10. Световое табло состоит из лампочек, каждая из которых может находиться в трёх состояниях («включено», «выключено» или «мигает»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 300 различных сигналов?

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n = 0 THEN t = 1 ELSE t = F(n - 1) * n PRINT t, END IF F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=0 то t:=1 иначе t:=F(n-1)*(n+1) вывод t, ' ' все знач:=t кон </pre>

Паскаль	Си
<pre>function F(n:integer):integer; var t:integer; begin if n = 0 then t:=1 else begin t:=F(n-1)*n write(t, ' '); end; F:=t end;</pre>	<pre>int F(int n) { int t; if (n == 0) t=1; else { t=F(n-1)*n; printf("%d ", t); } return t; }</pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(4)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 128.138.29.16

Маска: 255.255.252.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
16	34	124	128	138	28	0	255

Пример. Пусть искомый IP-адрес 192.168.128.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Всем сотрудникам фирмы для работы в локальной сети выдаётся идентификатор, состоящий из 9 символов. Первые 6 символов состоят из букв латинского алфавита (всего используется 8 прописных букв). Следующие три символа — три десятичные цифры от 0 до 9. Под хранение каждого такого идентификатора на компьютере отводится минимально возможное и одинаковое целое количество байт, при этом используется посимвольное

кодирование и все символы кодируются одинаковым и минимально возможным количеством битов. Сколько идентификаторов доступно для использования, если для их хранения достаточно 10 Кб.

14. Исполнитель *Черепашка* ползёт по клеткам бесконечной вертикальной клетчатой доски.

У исполнителя существует 4 команды:

Вправо n , Влево n , Вверх n , Вниз n (где n — целое число). При выполнении каждой из команд *Черепашка* перемещается на n шагов в в указанном направлении.

Цикл

ПОВТОРИ *число* РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что *последовательность команд* будет выполнена указанное *число* раз (число должно быть натуральным).

Черепашке был дан для исполнения следующий алгоритм (буквами n , a , b обозначены неизвестные числа, при этом $n > 1$):

вправо вниз вправо вниз вправо вверх влево вверх

НАЧАЛО

Вправо 25

ПОВТОРИ n РАЗ

Вниз 13

Вправо 54

Вниз a

Влево 36

Вверх 28

КОНЕЦ ПОВТОРИ

Вниз 11

КОНЕЦ

Укажите наименьшее возможное значение числа n , для которого найдётся такое значение числа a , что *Черепашка* после выполнения программы окажется на 5 клеток ниже исходной.

15. На рисунке 8 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

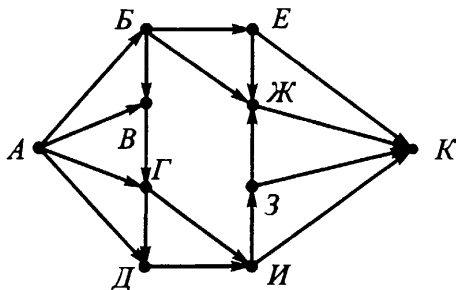


Рис. 8.

16. Сколько значащих нулей содержится в двоичной записи числа, которое можно представить в виде $16^{120} + 2^{800} - 4$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Андроид	235
Программа	415
Игры	225
Андроид & Игры	0
Андроид & Программа	80
Игры & Программа	95

Сколько страниц (в сотнях тысяч) будет найдено по запросу
Андроид | Игры | Программа?

18. На числовой прямой даны два отрезка: $P = [13, 54]$ и $Q = [7, 28]$. Укажите наименьшую возможную длину такого отрезка A , что логическое выражение

$$(\neg(x \in A) \rightarrow ((x \in P) \wedge (x \in A))) \vee ((x \in Q) \rightarrow (x \in P))$$

тождественно истинно, то есть принимает значение 1 при любом значении переменной x .

19. В программе обрабатывается одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8; 5; -11; 4; -6; -9; 4; -10; 7; 8 соответственно, т.е. $A[0] = 8$; $A[1] = 5$ и т.д.

Определите значение переменной k после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> k = 0 FOR i = 1 TO 9 IF A(i)>0 THEN A(i) = A(i-1) ELSE A(i) = i END IF k = k + A(i) NEXT i </pre>	<pre> k:=0 нц для i от 1 до 9 если A[i]>0 то A[i]:=A[i-1] иначе A[i]:=i все k:=k+A[i] кц </pre>
Паскаль	Си
<pre> k:=0; for i:=1 to 9 do begin if A[i]>0 then A[i]:=A[i-1] else A[i]:=i; k:=k+A[i] end </pre>	<pre> k=0; for (i=1; i<=9; i++) { if (A[i]>0) A[i]=A[i-1]; else A[i]=i; k+=A[i]; } </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наименьшее из таких чисел x , при вводе которого алгоритм печатает сначала 21, а потом 3.

Алгоритмический язык	Си
<pre> алг нач цел x, a, b, k ввод x a := 0 ; b := 10 нц пока x > 0 k := mod(x, 10); a := a + k если k < b то b := k все x := div(x, 10) кц вывод a, k, b кон </pre>	<pre> #include <stdio.h> void main() { int x, a, b, k; scanf("%d", &x); a = 0; b = 10; while (x > 0) { k = x % 10; a += k; if (k < b) b=k; x /= 10; } printf("%d \n %d", a, b); } </pre>

Бейсик	Паскаль
<pre> DIM X AS INTEGER DIM A AS INTEGER DIM B AS INTEGER DIM K AS INTEGER INPUT X A = 0 B = 10 WHILE X > 0 K = X MOD 10 A = A + K IF K < B THEN B = K X = X \ 10 WEND PRINT A, B </pre>	<pre> var x, a, b, k : integer; begin readln(x); a := 0; b := 10; while x > 0 do begin k := x mod 10; a := a + k; if k < b then b := k; x := x div 10 end; writeln(a); writeln(b); end. </pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

Алгоритмический язык	Си
<pre> алг нач цел a,b,s,K,L a:=-16; b:=16 K:=a; L:=F(a) нц для s от a до b если F(s) > L то K:=s L:=F(s) все кц вывод K кон алг цел F(цел x) нач знач := 5*(6-x)*(x+4) кон </pre>	<pre> #include<stdio.h> int F(int x) { return (5*(6-x)*(x+4)); } void main() { int a,b,s,K,L; a=-16; b:=16; K=a; L=F(a); for (s=a; s<=b; s++) if (F(s)>L) { K=s; L=F(s); } printf("%d", K); } </pre>

Бейсик	Паскаль
<pre> A = -16 : B = 16 K = A : L = F(A) FOR S = A TO B IF F(S) > L THEN K = S L = F(S) END IF NEXT S PRINT K FUNCTION F (X) F = 5 * (6 - X) * (X + 4) END FUNCTION </pre>	<pre> var a,b,s,K,L: integer; Function F(x:integer):integer; begin F:=5*(6-x)*(x+4); end; BEGIN a:=-16; b:=16; K:=a; L:=F(a); for s:=a to b do if F(s) > L then begin K:=s; L:=F(s) end; writeln(K) END. </pre>

22. У исполнителя X135 три команды, которым присвоены номера:

1. прибавь 1,
2. прибавь 3,
3. умножь на 5.

Первая из них увеличивает число на экране на 1, вторая — на 3, а третья — в 5 раз. Программа для исполнителя X135 — это последовательность команд. Сколько есть программ, которые число 1 преобразуют в число 14?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_{10}$, которые удовлетворяют всем перечисленным ниже условиям?

$$\begin{aligned}
 &\neg(x_1 \equiv x_2) \wedge ((x_1 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3)) = 0, \\
 &\neg(x_2 \equiv x_3) \wedge ((x_2 \wedge \neg x_4) \vee (\neg x_2 \wedge x_4)) = 0, \\
 &\dots \\
 &\neg(x_8 \equiv x_9) \wedge ((x_8 \wedge \neg x_{10}) \vee (\neg x_8 \wedge x_{10})) = 0.
 \end{aligned}$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_{10}$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число N , не превосходящее 10^9 . Если число чётное, то выводится сумма первой и последней цифр числа, в противном случае выводится первая цифра числа. Программист написал программу неправильно.

Алгоритмический язык

```
алг
нач
  цел N, S
  S:=0
  ввод N
  если mod(N,2)=0 то
    S:= mod(N,10)
    нц пока N > 0
      N:=div(N,10)
    кц
  все
  вывод S+N
кон
```

Паскаль

```
var N: longint;
    S: integer;
begin
  S:=0;
  readln(N);
  if N mod 2 = 0 then
    begin
      S:= N mod 10;
      while N > 0 do N:= N div 10;
    end;
  writeln(S+N)
end.
```

Бейсик
<pre>DIM N AS LONG S = 0 INPUT N IF N MOD 2 = 0 THEN S = N MOD 10 WHILE N > 0 N = N \ 10 WEND END IF PRINT S+N END</pre>
Си
<pre>#include <stdio.h> void main() { long N; int S; S = 0; scanf("%ld", &N); if N % 2 == 0 { S = N % 10; while (N>0) N /= 10; } printf("%d", S+N); }</pre>

Последовательно выполните следующее:

1. Напишите, что выведет программа при вводе каждого из чисел 615 и 214.

2. Приведите пример такого числа, при вводе которого программа, несмотря на ошибки, печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, — приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан массив из 30 элементов. Каждый элемент может принимать целочисленные значения от 0 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, который позволит найти сумму двузначных элементов массива, сумма цифр которых меньше 10. Гарантируется, что в исходном массиве есть хотя бы один такой элемент. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них. Исходные данные всегда подобраны так, что результат суммы не выходит за пределы объявленных типов данных.

Паскаль	Бейсик
<pre>const N=30; var a: array [0..N-1] of integer; i,k: integer; sum: longint; begin for i:=0 to N-1 do readln(a[i]); ... end.</pre>	<pre>CONST N = 30 DIM A(N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM SUM AS LONG FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>
Си	Естественный язык
<pre>#include <stdio.h> #define N 30 void main(void){ int a[N]; int i, k; long sum; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 30 элементов. Объявляем целочисленные переменные I, K и SUM. В цикле от 1 до 30 вводим элементы массива А с 1-го по 30-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться

на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого вами языка программирования.

26. Два игрока, Вова и Миша, играют в рамках соревнования в следующую игру. Перед ними лежат две кучки камней по 7 штук в каждой. У каждого игрока неограниченно много камней. Игроки ходят по очереди, при этом первый ход делает Миша. Ход состоит в том, что игрок докладывает в любую либо в каждую из куч два камня или же докладывает в каждую кучу по 1 камню. Игра завершается в тот момент, когда количество камней в одной кучке делится на 3, а в другой — на 2. Если при этом количество камней в обеих кучках в сумме чётно, то выигрывает Миша, в противном случае — Вова. Если количество камней в обеих кучках одинаково, то проигравший игрок не выбывает из дальнейших соревнований. Кто выигрывает при безошибочной игре обоих игроков? Выбывает ли из дальнейших соревнований проигравший игрок? Какими должны быть ходы игроков, если победитель выигрывает за минимальное число ходов? Ответ обоснуйте.

27. На вход программы подаются сведения о набранных на ЕГЭ баллах учениками данной школы по трём предметам. В первой строке сообщается количество учащихся N ($N \leq 100$), каждая из следующих N строк имеет формат: <Фамилия> <Инициалы> <БаллыПоРусскомуЯзыку> <БаллыПоМатематике> <БаллыПоИнформатике>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Инициалы> — строка, состоящая из 4 символов (буква, точка, буква, точка), <БаллыПоРусскомуЯзыку>, <БаллыПоМатематике>, <БаллыПоИнформатике> — целые числа в диапазоне от 0 до 100. Все элементы одной строки отделены друг от друга пробелом.

Пример входной строки:

Петров С.Н. 78 82 70

Требуется написать как можно более эффективную программу, которая будет выводить на экран фамилии и инициалы учеников, набравших максимальную сумму баллов по трём предметам (таких учеников может быть несколько), а также набранную ими сумму баллов.

Вариант № 6

Часть 1

1. Для кодирования некоторой последовательности, состоящей из букв русского алфавита, применили алгоритм шифра Вижинера:

1) Символы исходного алфавита нумеруются, начиная с нуля:

А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
20	21	22	23	24	25	26	27	28	29	30	31	32

2) Задаётся ключ кодирования — слово в исходном алфавите, например, МЕЛ.

3) Выписываются числа, соответствующие буквам кодируемого сообщения. Например, для сообщения КОМПЬЮТЕР получим последовательность чисел:

11 15 13 16 29 31 19 5 17

4) Под каждым числом полученной последовательности выписываются числа символов ключевого слова (в нашем случае, МЕЛ — 13 5 12), повторяющиеся до тех пор, пока не закончатся числа кодируемого слова :

11 15 13 16 29 31 19 5 17

13 5 12 13 5 12 13 5 12

5) В каждом столбце находят остаток от деления суммы его чисел на 33 — число символов исходного алфавита (для русского алфавита — 33).

Суммы чисел 24 20 25 29 34 43 32 10 29

Остатки от деления сумм на 33 24 20 25 29 1 10 32 10 29

6) Полученный числовой ряд преобразуется в символы исходного алфавита:

Ч У Ш Ъ Б Й Я Ъ

В результате кодирования некоторого сообщения с помощью данного алгоритма была получена последовательность символов МХЫЬЗЩЭ. В качестве ключа было использовано слово ЭХО. Укажите закодированное сообщение.

1) ВИЗИТКА 2) ДОГАДКА 3) ПАМЯТКА 4) МАРАФОН

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	F
1	0	1	1	0	1	1
1	1	1	0	0	1	1
1	1	1	0	0	0	1

Каким выражением может быть F ?

- 1) $\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4 \vee x_5 \vee x_6$
- 2) $x_1 \vee x_2 \vee x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6$
- 3) $\neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge x_4 \wedge x_5 \wedge x_6$
- 4) $x_1 \wedge x_2 \wedge x_3 \wedge \neg x_4 \wedge \neg x_5 \wedge \neg x_6$

3. Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы: символ «?» (вопросительный знак) означает ровно один произвольный символ; символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

В каталоге находятся шесть файлов:

sql.h
sylfaen.fch
sysvlp.sh
caseload.html
mplshlib.h
casual.http

Определите, по какой из масок из них будет отобрана указанная группа файлов:

sql.h
caseload.html
mplshlib.h

- 1) $*s*!?.*h*$
- 2) $?s?!*.*h?$
- 3) $*s?!*.*h*$
- 4) $*s*!*.h?$

4. Как представляется число 100_{10} в двоичной системе счисления?

5. На рисунке 9 изображена схема дорог между населёнными пунктами и обозначена стоимость перевозки пассажиров между соседними населёнными пунктами. Определите наименьшую стоимость перевозки из пункта A в пункт B .

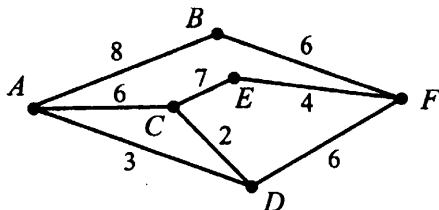


Рис. 9.

6. Дан алгоритм преобразования одного двузначного числа в другое.

1. Найти произведение цифр заданного числа и приписать её в конец числа.

2. В полученном числе оставить последние две цифры, остальные отбросить.

3. Если полученное число меньше 10 — прибавить 10.

Укажите наибольшее число, из которого в результате выполнения алгоритма будет получено число 10.

7. Дан фрагмент электронной таблицы.

	A	B	C	D
1	1	8	=МАКС(B1:B4)	10
2	2	4		11
3	3	9		12
4	4	5		13

Значения ячеек D1:D4 были получены следующим образом: в ячейку D4 записали формулу, затем скопировали её в ячейки диапазона D1:D3.

Определите, какая формула была записана в ячейку D4.

1) = A1 + C\$1 2) = A\$1 + C\$1 3) = A\$4 + C1 4) = A4 + C\$1

8. Определите значение переменной k после выполнения фрагмента программы, записанной ниже на различных языках программирования.

Паскаль	Си
<pre>s := 150; k := 1; while s > 0 do begin s := s - 2 * k; if s > 0 then k := k + 1 end; writeln(k);</pre>	<pre>s = 150; k = 1; while (s > 0) { s -= 2 * k; if (s > 0) k += 1; } printf("%d", k);</pre>

Бейсик	Алгоритмический язык
<pre> S = 150 K = 1 WHILE S > 0 S = S - 2 * K IF S > 0 THEN K = K + 1 END IF WEND PRINT K </pre>	<pre> s := 150 k := 1 нц пока s > 0 s := s - 2 * k если s > 0 то k := k + 1 все кц вывод k </pre>

9. Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 64 кГц и 32-битным разрешением. Запись длится 2 минуты, её результаты записываются в файл, сжатие данных не производится. Определите приблизительно размер полученного файла (в Мбайтах).

В качестве ответа укажите ближайшее к размеру полученного файла целое число, кратное 10.

10. Световое табло состоит из лампочек, каждая из которых может находиться в трёх состояниях («включено», «выключено» или «мигает»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 100 различных сигналов?

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Паскаль	Си
<pre> function F(n:integer):integer; var t:integer; begin if n = 0 then t:=1 else begin t:=F(n-1)+n write(t, ' '); end; F:=t end; </pre>	<pre> int F(int n) { int t; if (n == 0) t=1; else { t=F(n-1)+n; printf("%d ", t); } return t; } </pre>

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n = 0 THEN t = 1 ELSE t = F(n - 1) + n PRINT t, END IF F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=0 то t:=1 иначе t:=F(n-1)+(n+1) вывод t, ' ' все знач:=t кон </pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(6)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 156.148.124.64

Маска: 255.255.254.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
64	120	124	148	156	248	252	0

Пример. Пусть искомый IP-адрес 192.168.128.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Всем сотрудникам фирмы для работы в локальной сети выдаётся идентификатор, состоящий из 12 символов. Первые 3 символа состоят из букв латинского алфавита (всего используется 20 прописных букв). Следующие 9 символов — десятичные цифры от 0 до 9. Под хранение каждого такого идентификатора на компьютере отводится минимально возможное

и одинаковое целое количество байт, при этом используется посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством битов. Сколько идентификаторов доступно для использования, если для их хранения достаточно 14 Кб.

14. Исполнитель *Черепашка* ползёт по клеткам бесконечной вертикальной клетчатой доски.

У исполнителя существует 4 команды:

Вправо n , Влево n , Вверх n , Вниз n (где n — целое число). При выполнении каждой из команд *Черепашка* перемещается на n шагов в указанном направлении.

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

означает, что последовательность команд будет выполнена указанное число раз (число должно быть натуральным).

Черепашке был дан для исполнения следующий алгоритм (буквами n , a , b обозначены неизвестные числа, при этом $n > 1$):

вправо вниз вправо вниз вправо вверх влево вверх

НАЧАЛО

Влево 65

ПОВТОРИ n РАЗ

Вниз 24

Вправо 14

Вниз a

Вверх 56

Влево 48

КОНЕЦ ПОВТОРИ

Вниз 15

КОНЕЦ

Укажите наименьшее возможное значение числа n , для которого найдётся такое значение числа a , что *Черепашка* после выполнения программы окажется на 12 клеток правее исходной.

15. На рисунке 10 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

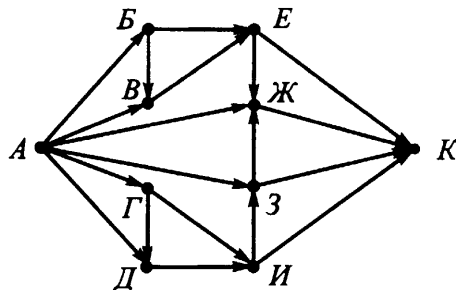


Рис. 10.

16. Сколько значащих нулей содержится в двоичной записи числа, которое можно представить в виде $8^{510} + 4^{1500} - 16$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$. Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Праздники	520
Выходные	715
2015	612
Праздники 2015	1100
Праздники Выходные	1235
2015 Выходные	1200

Сколько страниц (в сотнях тысяч) будет найдено по запросу

(Праздники | 2015) & Выходные?

18. На числовой прямой даны два отрезка: $P = [17, 44]$ и $Q = [10, 32]$. Укажите наименьшую возможную длину такого отрезка A , что логическое выражение

$$((x \in A) \rightarrow \neg((x \in P) \vee (x \in A))) \vee \neg((x \in Q) \rightarrow (x \in P))$$

тождественно истинно, то есть принимает значение 1 при любом значении переменной x .

19. В программе обрабатывается одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8; 5; -11; 4; -6; -9; 4; -10; 7; 8 соответственно, т.е. $A[0] = 8$; $A[1] = 5$ и т.д.

Определите значение переменной k после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> k = 0 FOR i = 0 TO 8 IF A(i)<0 THEN A(i) = A(i+1) ELSE A(i) = i END IF k = k + A(i) NEXT i </pre>	<pre> k:=0 нц для i от 0 до 8 если A[i]<0 то A[i]:=A[i+1] иначе A[i]:=i все k:=k+A[i] кц </pre>
Паскаль	Си
<pre> k:=0; for i:=0 to 8 do begin if A[i]<0 then A[i]:=A[i+1] else A[i]:=i; k:=k+A[i] end </pre>	<pre> k=0; for (i=0; i<9; i++){ if (A[i]<0) A[i]=A[i+1]; else A[i]=i; k+=A[i]; } </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наибольшее четырёхзначное число x , при вводе которого алгоритм печатает сначала 17, а потом 6.

Бейсик	Паскаль
<pre> INPUT X A = 0 B = 0 WHILE X > 0 K = X MOD 10 A = A + K IF K > B THEN B = K X = X \ 10 WEND PRINT A, B </pre>	<pre> var x, a, b, k : integer; begin readln(x); a := 0; b := 0; while x > 0 do begin k := x mod 10; a := a + k; if k > b then b := k; x := x div 10 end; writeln(a); writeln(b); end. </pre>

Алгоритмический язык	Си
алг нач цел x, a, b, k ввод x a := 0 b := 0 нц пока x > 0 k := mod(x, 10) a := a + k если k > b то b := k все x := div(x, 10) кц вывод a, b кон	#include <stdio.h> void main() { int x, a, b, k; scanf("%d", &x); a = 0; b = 0; while (x > 0) { k = x % 10; a += k; if (k > b) b=k; x /= 10; } printf("%d \n %d", a, b); }

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

Алгоритмический язык	Си
алг нач цел a,b,s,K,L a:=-14; b:=14 K:=a; L:=F(a) нц для s от a до b если F(s) > L то K:=s L:=F(s) все кц вывод K кон алг цел F(цел x) нач знач := 3*(x-5)*(2-x) кон	#include<stdio.h> int F(int x) { return (3*(x-5)*(2-x)); } void main() { int a,b,s,K,L; a=-14; b=14; K=a; L=F(a); for (s=a; s<=b; s++) if (F(s)>L) { K=s; L=F(s); } printf("%d", K); }

Бейсик	Паскаль
<pre> A = -14 : B = 14 K = A : L = F(A) FOR S = A TO B IF F(S) > L THEN K = S L = F(S) END IF NEXT S PRINT K FUNCTION F (X) F = 3 * (X - 5) * (2 - X) END FUNCTION </pre>	<pre> var a,b,s,K,L: integer; Function F(x:integer):integer; begin F:=3*(x-5)*(2-x); end; BEGIN a:=-14; b:=14; K:=a; L:=F(a); for s:=a to b do if F(s) > L then begin K:=s; L:=F(s) end; writeln(K) END. </pre>

22. У исполнителя X123 три команды, которым присвоены номера:

1. прибавь 1,
2. прибавь 2,
3. умножь на 3.

Первая из них увеличивает число на экране на 1, вторая — на 2, а третья — в 3 раза. Программа для исполнителя X123 — это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 10?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$\neg(x_1 \equiv x_2) \wedge (x_1 \vee x_3) \wedge (\neg x_1 \vee \neg x_3) = 0,$$

$$\neg(x_2 \equiv x_3) \wedge (x_2 \vee x_4) \vee (\neg x_2 \vee \neg x_4) = 0,$$

...

$$\neg(x_6 \equiv x_7) \wedge (x_6 \vee x_8) \vee (\neg x_6 \vee \neg x_8) = 0.$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число N , не превосходящее 10^9 , и выводится сумма цифр этого числа, не превосходящих 5. Программист написал программу неправильно.

Алгоритмический язык

```
алг
нач
  цел N, VAL, S
  ввод N
  S:=0
  нц пока N > 0
    VAL:= mod(N,10)
    если VAL < 5 то
      S:=VAL
    все
    N:=div(N,10)
  кц
  вывод S
кон
```

Паскаль

```
var N: longint;
    S, VAL: integer;
begin
  readln(N);
  S:=0;
  while N > 0 do
  begin
    VAL:= N mod 10;
    if VAL < 5 then
      S:= VAL;
    N:= N div 10
  end;
  writeln(S)
end.
```

Бейсик
<pre>DIM N AS LONG INPUT N S = 0 WHILE N > 0 VAL = N MOD 10 IF VAL < 5 THEN S=VAL N = N \ 10 WEND PRINT S END</pre>
Си
<pre>#include <stdio.h> void main() { long N; int VAL, S; scanf("%ld", &N); S = 0; while (N>0) { VAL = N % 10; if (VAL < 5) S = VAL; N /= 10; } printf("%d", S); }</pre>

Последовательно выполните следующее.

1. Напишите, что выведет программа при вводе числа 4715.
2. Приведите пример такого числа, что, несмотря на ошибки, программа печатает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, — приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования. Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую дру-

гой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан массив из 50 элементов. Каждый элемент может принимать целочисленные значения от -500 до 500 . Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий определить, на сколько число положительных элементов массива больше, чем число отрицательных, и вывести найденное значение. Гарантируется, что в исходном массиве есть и отрицательные, и положительные элементы, и положительных элементов больше. Исходные данные объявлены так, как показано ниже.

Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Бейсик
<pre>const N=50; var a: array [1..N] of integer; i,kol_p,kol_m: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>CONST N = 50 DIM A(N) AS INTEGER DIM I AS INTEGER DIM KOL_P AS INTEGER DIM KOL_M AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>
Си	Естественный язык
<pre>#include <stdio.h> #define N 50 void main(void){ int a[N]; int i, kol_p, kol_m; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 50 элементов. Объявляем целочисленные переменные I, KOL_P и KOL_M. В цикле от 1 до 50 вводим элементы массива А с 1-го по 50-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы.

В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого вами языка программирования.

26. Два игрока, Георгий и Валерий, играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых 5, а во второй — 7 камней. У каждого игрока неограниченно много камней.

Игроки ходят по очереди, при этом первый ход делает Георгий. Ход состоит в том, что игрок докладывает в одну из кучек либо два, либо четыре камня. Игра завершается в тот момент, когда количество камней в обеих кучках будет не меньше 15, при этом в одной из кучек будет на 4 камня больше, чем в другой.

Побеждает игрок, ход которого оказался последним, при этом проигравший игрок предоставляет победителю приз в виде пирожков, количество которых равно сумме камней в кучках, это количество умножается на 2, если в первой кучке камней оказалось больше.

Кто выигрывает при безошибочной игре обоих игроков? Сколько пирожков он получает? Какими должны быть ходы игроков при выигрыше за минимальное число ходов? Ответ обоснуйте.

27. На вход программы подаются сведения об N (не более 100) пользователях сети. Первая строка содержит адрес сети и подсети в формате xxx , либо $xxx.xxx$, либо $xxx.xxx.xxx$ (везде далее под xxx понимается целое число в диапазоне от 0 до 255). Во второй строке задаётся число N . Каждая из следующих N строк имеет формат: <фамилия> <логин> <IP-адрес>, причём элементы строки разделены пробелами, и IP-адрес имеет вид $xxx.xxx.xxx.xxx$ (например, Еремеев егет 192.168.0.1). Требуется написать программу, которая выводит на экран в алфавитном порядке логины тех пользователей, чьи компьютеры находятся в заданной подсети, а также IP-адреса их компьютеров.

Вариант № 7

Часть 1

1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, решили использовать неравномерный двоичный код, позво-

ляющий однозначно декодировать двоичную последовательность, появляющуюся на приёмной стороне канала связи. Использовали код: А – 10, Б – 111, В – 011, Г – 01. Укажите, каким кодовым словом может быть закодирована буква Д. Код должен удовлетворять свойству однозначного декодирования.

- 1) 11 2) 100 3) 110 4) 101

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	x_7	F
1	0	1	0	1	1	1	1
1	1	1	0	0	0	0	1
0	1	0	1	1	1	1	0

Каким из приведённых ниже выражений может быть F ?

- 1) $\neg(x_1 \vee \neg x_2 \vee x_3) \wedge \neg(x_4 \vee x_5 \vee x_6 \vee x_7)$
 2) $(x_1 \vee x_2 \vee x_3) \wedge (\neg x_4 \vee x_5 \vee x_6 \vee \neg x_7)$
 3) $(x_1 \wedge \neg x_2 \wedge x_3) \vee \neg(x_4 \wedge x_5 \wedge x_6 \wedge x_7)$
 4) $(x_1 \wedge x_2 \wedge x_3) \vee (\neg x_4 \wedge x_5 \wedge x_6 \wedge \neg x_7)$

3. Ниже приведены фрагменты таблиц базы данных, содержащей информацию о расписании занятий.

ID_D	Класс
100	9 А
103	9 Б
105	10 А
131	10 Б
207	11 А
209	11 Б

ID_D	Преподаватель	Время
100	Сафронов И. О.	10 : 15 – 11 : 00
105	Петров Т. Д.	11 : 20 – 12 : 05
131	Сафронов И. О.	13 : 25 – 14 : 10
103	Сафронов И. О.	11 : 20 – 12 : 05
100	Никитина Л. Ю.	14 : 25 – 15 : 25
209	Сафронов И. О.	12 : 20 – 13 : 05
207	Петров Т. Д.	13 : 25 – 14 : 10

По данным этих таблиц определите, в каком классе преподаватель Сафронов И. О. проводит последний урок?

- 1) 9 А 2) 10 Б 3) 11 А 4) 11 Б

4. Укажите число байт, необходимое для записи числа 2^{75} ?

5. На рисунке 11 изображена схема дорог между населёнными пунктами и обозначена стоимость перевозки пассажиров между соседними населён-

ными пунктами. Определите наименьшую стоимость перевозки из пункта А в пункт В.

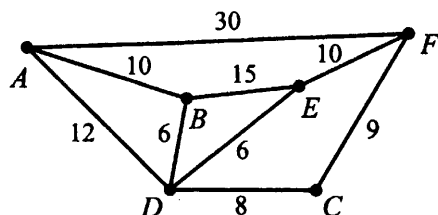


Рис. 11.

6. У исполнителя имеется две команды:

- 1) **Вычти 3,**
- 2) **Умножь на 5.**

Первая команда уменьшает число на экране на 3, вторая — увеличивает его в 5 раз.

Запишите порядок команд в программе преобразования числа 6 в число 117, содержащей не более 5 команд, указывая лишь номера команд. Например 1122 — это программа: **вычти 3, вычти 3, умножь на 5, умножь на 5**, которая преобразует число 7 в число 25.

7. Дан фрагмент электронной таблицы в режиме отображения формул.

	A	B	C
1	2	5	
2	$= A1 * (B1 + 7)$	$= (B1 + C1) * 3$	$= 2 * A1 * (1 + B1)$

Какое целое число должно быть записано в ячейке C1, чтобы построенная после выполнения вычислений диаграмма по значениям диапазона ячеек A2 : C2 соответствовала рисунку 12?



Рис. 12.

Известно, что все значения диапазона, по которым построена диаграмма, имеют один и тот же знак.

8. Определите, что будет напечатано в результате работы следующего фрагмента программы, записанного ниже на различных языках программирования (переменные m , n , i — целого типа).

Алгоритмический язык
<pre>м := 0 п := 1 нц для i от 1 до 5 п := 3 * п - i если п > 10 то м := п + м иначе м := м + 1 все кц вывод м</pre>
Бейсик
<pre>M = 0 N = 1 FOR I = 1 TO 5 N = 3 * N - I IF N > 10 THEN M = N + M ELSE M = M + 1 NEXT I PRINT M</pre>
Паскаль
<pre>м := 0; п := 1; for i := 1 to 5 do begin п := 3 * п - i; if п > 10 then м := п + м else м := м + 1 end; write(m);</pre>

Си

```
m = 0; n = 1;
for (i = 1; i <= 5; i++) {
    n = 3 * n - i;
    if (n > 10) m = n + m;
    else m = m + 1;
}
printf("%d ", m);
```

9. Документ объёмом 16 Мбайт можно передать с одного компьютера на другой двумя способами.

А. Сжать архиватором, передать архив по каналу связи, распаковать.

Б. Передать по каналу связи без использования архиватора.

Какой способ быстрее и насколько, если:

— средняя скорость передачи данных по каналу связи составляет 2^{22} бита в секунду;

— объём сжатого архиватором документа равен 25% исходного;

— время, требуемое на сжатие документа, — 15 секунд, на распаковку — 4 секунды?

В ответе напишите букву А, если быстрее способ А, или Б, если быстрее способ Б. Сразу после буквы напишите число, обозначающее, на сколько секунд один способ быстрее другого. Так, например, если способ Б быстрее способа А на 13 секунд, в ответе нужно написать Б13. Единицы измерения «секунд», «сек.», «с.» к ответу добавлять не нужно.

10. Все пятибуквенные слова, составленные из букв В, Е, М, Р, Я, записаны в алфавитном порядке и пронумерованы. Вот начало списка:

1. ВВВВВ
2. ВВВВЕ
3. ВВВВМ
4. ВВВВР
5. ВВВВЯ
6. ВВВЕВ

...

Под каким номером стоит слово ВРЕМЯ?

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n = 1 THEN t = 1 ELSE t = F(n - 1) + 2*n PRINT t, END IF F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=1 то t:=1 иначе t:=F(n-1)+2*(n+1) вывод t, ' ' все знач:=t кон </pre>
Паскаль	Си
<pre> function F(n:integer):integer; var t:integer; begin if n = 1 then t:=1 else begin t:=F(n-1)+2*n write(t, ' '); end; F:=t end; </pre>	<pre> int F(int n) { int t; if (n == 1) t=1; else { t=F(n-1)+2*n; printf("%d ", t); } return t; } </pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(8)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 172.156.131.46

Маска: 255.255.128.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
156	46	131	128	172	259	0	255

Пример. Пусть искомый IP-адрес 192.168.128.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Каждый пользователь сети обладает паролем вида XZYYYYYZ, где X — строчная буква английского алфавита (всего используется 26 различных символов), Y — цифра 0, 1, ..., 9 и Z — либо 0, либо 1. Кодирование паролей осуществляется посимвольно. Каждый из символов X, Y и Z кодируется минимально возможным количеством бит, а каждый пароль занимает одинаковое целое число байт. Определите объём памяти (в байтах), необходимый для хранения 20 паролей. В ответе запишите только число, слово «байт» писать не нужно.

14. Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости: **вверх**, **вниз**, **влево** и **вправо**. При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх, вниз, влево, вправо. Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

слева свободно, справа свободно, сверху свободно, снизу свободно.

Цикл

ПОКА <условие> команда

выполняется, пока условие истинно, иначе происходит переход на следующую строку. Если РОБОТ начнёт движение в сторону стены, то он разрушится и выполнение программы прервётся.

Сколько клеток приведённого лабиринта (см. рис. 13) соответствуют требованию: выполнив предложенную ниже программу, РОБОТ уцелеет и остановится в той же клетке, с которой он начал движение?

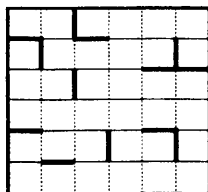


Рис. 13.

НАЧАЛО

ПОКА <сверху свободно> вправо

ПОКА <справа свободно> вниз

ПОКА <снизу свободно> влево

ПОКА <слева свободно> вверх

КОНЕЦ

15. На рисунке 14 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?

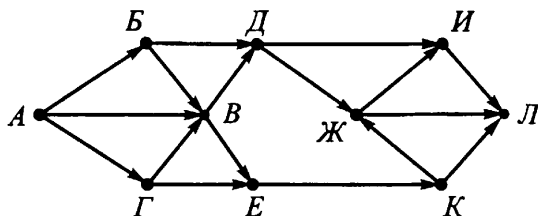


Рис. 14.

16. Сколько единиц содержится в двоичной записи значения выражения: $16^{700} - 2^{1100} + 9$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&».

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Электронное	460
Приложение	910
Учебник	560
Электронное Учебник	600
Электронное & Приложение	0
Учебник & Приложение	0

Сколько страниц будет найдено по запросу
(Электронное & Учебник) | Приложение?

18. Сколько натуральных чисел A удовлетворяют логическому выражению

$$(A \geq 8) \wedge \neg(((A > 15) \vee (A < 10)) \rightarrow (A > 20))?$$

19. В программе используется одномерный числовой массив A с индексами от 0 до 9. Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> S:=0 FOR I=0 TO 9 A(I)=I*5 NEXT I FOR I=0 TO 9 A(I)=A(I) MOD 5 S=S+A(I) NEXT I </pre>	<pre> S:=0 нц для i от 0 до 9 A[i]:=i*5 кц нц для i от 0 до 9 A[i]:= mod(A[i],5) S:= S+A[i] кц </pre>

Паскаль
<pre> S:=0; for i:=0 to 9 do a[i]:=i*5; for i:=0 to 9 do begin a[i]:=a[i] mod 5; S:=S+a[i] end; </pre>

Си
<pre> S=0; for (int i=0; i<10; i++) a[i]=i*5; for (int i=0; i<10; i++) { a[i]%=5; S+=a[i]; } </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает число A . Укажите такое число x , при вводе которого алгоритм печатает 14263.

Бейсик	Паскаль
<pre> INPUT X K = 1 WHILE X > 0 IF (X MOD 10) MOD 3 = 0 THEN K = K * 10 B = B * 10 + X MOD 10 ELSE A = A * 10 + X MOD 10 END IF X = X \ 10 WEND A = A * K + B PRINT A </pre>	<pre> var x, K, A, B: integer; begin readln(x); K := 1; while x > 0 do begin if (x mod 10) mod 3 = 0 then begin K := K * 10; B:= B*10 + x mod 10 end else A:= A*10 + x mod 10; x := x div 10 end; A := A * K + B; writeln(A) end. </pre>
Алгоритмический язык	Си
<pre> алг нач цел x, K, A, B ввод x K := 1 нц пока x > 0 если mod(mod(x,10),3)=0 то K := K*10 B := B*10 + mod(x,10) иначе A := A*10 + mod(x,10) все x := div(x,10) кц A := A * K + B вывод A кон </pre>	<pre> #include <stdio.h> void main() { int x, K, A, B; scanf ("%d", &x); K = 1; while (x > 0) { if ((x % 10) % 3 == 0) { K *= 10; B = B * 10 + x % 10;} else A = A * 10 + x % 10; x = x / 10; } A = A * K + B; printf("%d ", A); } </pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

Паскаль	Бейсик
<pre> var a,E: integer; Function F1(x:integer):integer; begin F1:=1-2*x end; Function F2(x:integer):integer; begin F2:=4-3*x end; BEGIN a:=-10; E:=1; while abs(F1(a)-F2(a))>E do a:=a+1; writeln(a) END.</pre>	<pre> A = -10 E = 1 WHILE ABS(F1(A)-F2(A))>E A = A+1 WEND NEXT A FUNCTION F1 (X) F1 = 1-2*X END FUNCTION FUNCTION F2 (X) F2 = 4-3*x END FUNCTION</pre>

Алгоритмический язык	Си
<pre> алг нач цел a,E a:=-10; E:=1 нц пока iabs(F1(a)-F2(a))>E a:=a+1 кц вывод a кон алг цел F1(цел x) знач := 1-2*x кон алг цел F2(цел x) нач знач := 4-3*x кон</pre>	<pre> #include<stdio.h> int F1(int x) { return (1-2*x); } int F2(int x) { return (4-3*x); } void main() { int a,E; a=-10; E:=1; while (abs(F1(a)-F2(a))>E) a++; printf("%d", a); }</pre>

22. У исполнителя 1X две команды, которым присвоены номера:

1. прибавь 1,

2. увеличь число десятков на 1.

Например: при помощи команды 2 число 48 преобразуется в 58. Если перед выполнением команды 2 вторая с конца цифра равна 9, она не изменяется. Программа для исполнителя 1X — это последовательность команд.

Определите количество программ, которые число 13 преобразуют в число 45?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_7, x_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \equiv \neg x_2) \vee (\neg x_1 \wedge x_3) \vee \neg(x_1 \equiv x_3) = 1$$

$$(x_2 \equiv \neg x_3) \vee (\neg x_2 \wedge x_4) \vee \neg(x_2 \equiv x_4) = 1$$

$$(x_3 \equiv \neg x_4) \vee (\neg x_3 \wedge x_5) \vee \neg(x_3 \equiv x_5) = 1$$

$$(x_4 \equiv \neg x_5) \vee (\neg x_4 \wedge x_6) \vee \neg(x_4 \equiv x_6) = 1$$

$$(x_6 \equiv \neg x_7) \vee (\neg x_6 \wedge x_8) \vee \neg(x_6 \equiv x_8) = 1$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, \dots, x_7, x_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число N , не превосходящее 10^9 .

Программа должна выводить на экран количество цифр, меньших 6, и минимальное из этих цифр. Если число не содержит цифр, меньших 6, то требуется на экран вывести «NO».

Программист написал программу неправильно.

Алгоритмический язык

алг

нач

цел N, K, VAL, MIN

ввод N

MIN:=10

K:=0

нц пока N > 0

VAL:= mod(N,10)

если VAL < 6 то

K:=K+VAL

если VAL < MIN то MIN:=VAL все

все

N:=div(N,10)

кц

если MIN > 0 то

вывод K, нс

вывод MIN

иначе

вывод "No"

кон

Бейсик

DIM N AS LONG

INPUT N

MIN = 10

K = 0

WHILE N > 0

VAL = N MOD 10

IF VAL < 6 THEN

K=K+VAL

IF VAL < MIN THEN MIN=VAL

END IF

N = N \ 10

WEND

IF MIN > 0 THEN

PRINT K

PRINT MIN

ELSE

PRINT "No"

END IF

END

Паскаль

```
var N: longint;  
    K, VAL, MIN: integer;  
begin  
    readln(N);  
    MIN:=0; K:=0;  
    while N > 0 do begin  
        VAL:= N mod 10;  
        if VAL < 6 then begin  
            K:= K+VAL;  
            if VAL < MIN then MIN:= VAL  
        end;  
        N:= N div 10  
    end;  
    if MIN > 0 then begin  
        writeln(K); writeln(MIN)  
    end  
    else writeln('No')  
end.
```

Си

```
#include <stdio.h>  
void main() {  
    long N;  
    int K, VAL, MIN;  
    scanf("%ld", &N);  
    MIN=0; K=0;  
    while (N>0) {  
        VAL = N % 10;  
        if (VAL < 6) {  
            K += VAL;  
            if (VAL < MIN) MIN = VAL;  
        }  
        N /= 10;  
    }  
    if (MIN > 0) {  
        printf("%d\n %d\n", K, MIN);  
    }  
    else printf("No");  
}
```

Последовательно выполните следующее.

1. Напишите, что выведет программа при вводе числа 4357.
2. Приведите пример такого числа, что, несмотря на ошибки, программа печатает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, — приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от 0 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести максимальное значение среди трёхзначных элементов массива, делящихся на 5. Если в исходном массиве нет трёхзначного элемента, кратного 5, то следует вывести сообщение «Не найдено». Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Бейсик
<pre>const N=20; var a: array [1..N] of integer; i,k,max: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>CONST N = 20 DIM A(N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM MAX AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>

Си	Естественный язык
<pre>#include <stdio.h> #define N 20 void main(void){ int a[N]; int i, k, max; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 20 элементов. Объявляем целочисленные переменные I, K и MAX. В цикле от 1 до 20 вводим элементы массива А с 1-го по 20-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае Вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого Вами языка программирования.

26. Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(-1, 1)$.

Ход состоит в том, что игрок перемещает фишку из точки с координатами (x, y) в одну из трёх точек: или в точку с координатами $(2x + 1, y + 2)$, или в точку с координатами $(x, y + 1)$, или в точку с координатами $(x, 2y)$.

Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами $(0, 0)$ не меньше 10 единиц.

Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

27. С клавиатуры вводится число записей N , не превышающее 1000, содержащих сведения о сетевом трафике, проходящем через компьютер, а затем сами записи в формате <время> <источник> <приёмник>, где <время> записано в виде xx:xx:xx, <источник> и <приёмник> в виде xxx.xxx.xxx.xxx (ip-адреса).

Например, 11:15:00 192.168.68.12 192.168.68.7.

Затем также с клавиатуры вводится запрос к этим записям в формате $\text{time}=\langle \text{время} \rangle \text{src}=\langle \text{источник} \rangle \text{dst}=\langle \text{приемник} \rangle$.

Причем в запросе любое из полей не обязательно, и если запросом является пустая строка, то должны выводиться все записи.

Например, $\text{time}=11:15:00 \text{dst}=192.168.68.7$.

Напишите программу, которая обрабатывает такой запрос и выводит все записи, удовлетворяющие ему.

Вариант № 8

Часть 1

1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, решили использовать неравномерный двоичный код, позволяющий однозначно декодировать двоичную последовательность, появляющуюся на приёмной стороне канала связи. Использовали код: А – 11, Б – 001, В – 010, Г – 100. Укажите, каким кодовым словом может быть закодирована буква Д. Код должен удовлетворять свойству однозначного декодирования.

1) 1

2) 110

3) 10

4) 111

2. Дан фрагмент таблицы истинности выражения F :

x_1	x_2	x_3	x_4	x_5	x_6	x_7	F
1	0	1	0	1	1	1	0
1	1	1	1	0	0	0	0
0	1	0	1	1	1	1	0

Каким из приведённых ниже выражений может быть F ?

1) $\neg(x_1 \vee \neg x_2 \vee x_3) \wedge (x_4 \vee x_5 \vee x_6 \vee x_7)$

2) $(x_1 \vee x_2 \vee x_3 \vee x_4) \wedge \neg(x_5 \vee \neg x_6 \vee x_7)$

3) $\neg(x_1 \wedge \neg x_2 \wedge x_3) \vee (x_4 \wedge x_5 \wedge x_6 \wedge x_7)$

4) $(x_1 \wedge x_2 \wedge x_3 \wedge x_4) \vee \neg(x_5 \wedge \neg x_6 \wedge x_7)$

3. Ниже приведены фрагменты таблиц базы данных, содержащей информацию о студентах и о посещаемых ими предметных курсах.

ID_S	Ф. И. О	Курс	Группа
335	Акимов С.Ю.	3	2
337	Бербин К.Н.	2	1
340	Волковая А.С.	4	1
341	Донецкий П.А.	4	2
344	Жилина Ю.Г.	3	1
345	Мамин Е.Н.	4	2
349	Оришко Г.И.	4	1

ID_S	Предмет
335	Физика
335	Информатика
340	Информатика
342	Правоведение
344	Правоведение
344	Информатика
345	Информатика
349	Базы данных

По данным этих таблиц определите, сколько студентов первой группы посещают предмет «информатика».

- 1) 1 2) 2 3) 3 4) 0

4. Укажите число байт, необходимое для записи числа 2^{65} ?

5. На рисунке 15 изображена схема дорог между населёнными пунктами и обозначена стоимость перевозки пассажиров между соседними населёнными пунктами. Определите наименьшую стоимость перевозки из пункта А в пункт В.

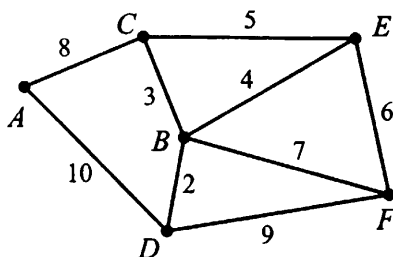


Рис. 15.

6. У исполнителя имеется две команды:

- 1) Вычти 4;
- 2) Умножь на 4.

Первая команда уменьшает число на экране на 4, вторая — увеличивает его в 4 раза.

Запишите порядок команд в программе преобразования числа 7 в число 16, содержащей не более 5 команд, указывая лишь номера команд. На-

пример 1121 — это программа:

вычти 4,
 вычти 4,
 умножь на 4,
 вычти 4,

которая преобразует число 13 в число 16.

7. Дан фрагмент электронной таблицы в режиме отображения формул.

	A	B	C
1		5	2
2	$= (A1 + 3)/C1$	$= A2 - B1$	$= A1 - (C1 + 4)$

Какое целое число должно быть записано в ячейке A1, чтобы построенная после выполнения вычислений диаграмма по значениям диапазона ячеек A2 : C2 соответствовала рисунку 16?



Рис. 16.

Известно, что все значения диапазона, по которым построена диаграмма, имеют один и тот же знак.

8. Определите, что будет напечатано в результате работы следующего фрагмента программы, записанного ниже на различных языках программирования (переменные m , n , i — целого типа).

Алгоритмический язык
<pre> m := 0; n := 3 нц для i от 1 до 4 n := n + 2*i если n > 10 то m := 2*n - m иначе m := m + 1 все кц вывод m </pre>

Бейсик
<pre>M = 0 : N = 3 FOR I = 1 TO 4 N = N + 2*I IF N > 10 THEN M = 2*N - M ELSE M = M + 1 NEXT I PRINT M</pre>
Паскаль
<pre>m := 0; n := 3; for i := 1 to 4 do begin n := n + 2*i; if n > 10 then m := 2*n - m else m := m + 1 end; write(m);</pre>
Си
<pre>m = 0; n = 3; for (i = 1; i <= 4; i++) { n = n + 2*i; if (n > 10) m = 2*n - m; else m = m + 1; } printf("%d ", m);</pre>

9. Документ объёмом 12 Мбайт можно передать с одного компьютера на другой двумя способами.

А. Сжать архиватором, передать архив по каналу связи, распаковать.

Б. Передать по каналу связи без использования архиватора.

Какой способ быстрее и насколько, если:

— средняя скорость передачи данных по каналу связи составляет 2^{24} бита в секунду;

— объём сжатого архиватором документа равен 50% исходного;

— время, требуемое на сжатие документа, — 14 секунд, на распаковку — 6 секунд?

В ответе напишите букву А, если быстрее способ А, или Б, если быстрее способ Б. Сразу после буквы напишите число, обозначающее, на сколько секунд один способ быстрее другого. Так, например, если способ Б быстрее способа А на 13 секунды, в ответе нужно написать Б13. Единицы измерения «секунд», «сек.», «с.» к ответу добавлять не нужно.

10. Все пятибуквенные слова, составленные из букв А, В, С, записаны в алфавитном порядке. Вот начало списка:

1. ААААА
2. ААААВ
3. ААААС
4. АААВВ

...

Определите номер строки, в которой будет находиться слово АССВВ.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n = 1 THEN t = 100 ELSE t = F(n - 1) - n PRINT t, END IF F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=1 то t:=100 иначе t:=F(n-1)-(n+1) вывод t, ' ' все знач:=t кон </pre>
Паскаль	Си
<pre> function F(n:integer):integer; var t:integer; begin if n = 1 then t:=100 else begin t:=F(n-1)-n write(t, ' '); end; F:=t end; </pre>	<pre> int F(int n) { int t; if (n == 1) t=100; else { t=F(n-1)-n; printf("%d ", t); } return t; } </pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(8)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 176.154.136.89

Маска: 255.255.192.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
144	176	255	136	0	128	89	154

Пример. Пусть искомый IP-адрес 192.168.128.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. Каждый пользователь сети обладает паролем вида $XYYYXZ$, где X — строчная буква английского алфавита (всего используется 26 различных символов), Y — цифра 0, 1, ..., 9 и Z — либо 0, либо 1. Кодирование паролей осуществляется посимвольно. Каждый из символов X , Y и Z кодируется минимально возможным количеством бит, а каждый пароль занимает одинаковое целое число байт. Определите объём памяти (в байтах), необходимый для хранения 18 паролей. В ответе запишите только число, слово «байт» писать не нужно.

14. Система команд исполнителя *РОБОТ*, «живущего» в прямоугольном лабиринте на клетчатой плоскости: **вверх**, **вниз**, **влево** и **вправо**. При выполнении любой из этих команд *РОБОТ* перемещается на одну клетку соответственно: вверх, вниз, влево, вправо. Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится *РОБОТ*:

слева свободно, справа свободно, сверху свободно, снизу свободно.

Цикл

ПОКА <условие> команда

выполняется, пока условие истинно, иначе происходит переход на следующую строку. Если **РОБОТ** начнёт движение в сторону стены, он разрушится, и программа прервётся.

Сколько клеток приведённого лабиринта (см. рис. 17) соответствуют требованию: выполнив предложенную ниже программу, **РОБОТ** уцелеет и остановится в той же клетке, с которой он начал движение?

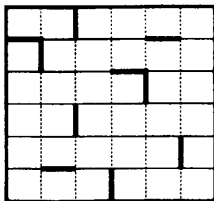


Рис. 17.

НАЧАЛО

ПОКА <слева свободно> вверх

ПОКА <сверху свободно> вправо

ПОКА <справа свободно> вниз

ПОКА <снизу свободно> влево

КОНЕЦ

15. На рисунке 18 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?

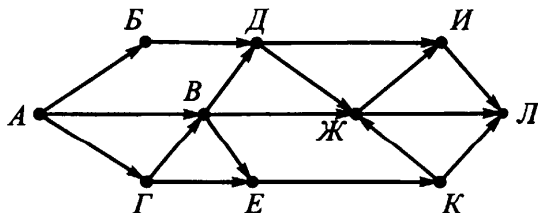


Рис. 18.

16. Сколько единиц содержится в двоичной записи значения выражения: $8^{1050} - 4^{1020} + 17$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&».

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Фантастика	390
Триллеры	540
Приключения	620
Приключения Фантастика	850
Триллеры & Приключения	120
Фантастика Триллеры	930

Сколько страниц будет найдено по запросу
Приключения & (Фантастика | Триллеры)?

18. Сколько натуральных чисел A не удовлетворяют логическому выражению

$$((A < 10) \rightarrow (A \leq 12)) \wedge ((A > 5) \rightarrow (A \geq 11))?$$

19. В программе используется одномерный числовой массив A с индексами от 0 до 9. Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> S=0 FOR I=0 TO 9 A(I)=10-I NEXT I FOR I=0 TO 4 A(I)=A(I+5) S=S+A(I) NEXT I </pre>	<pre> S:=0 нц для i от 0 до 9 A[i]:=10-i кц нц для i от 0 до 4 A[i]:=A[i+5] S:= S+A[i] кц </pre>

Паскаль	Си
<pre> S:=0; for i:=0 to 9 do a[i]:=10-i; for i:=0 to 4 do begin a[i]:=a[i+5]; S:=S+a[i] end;</pre>	<pre> S=0; for (int i=0; i<=9;i++) a[i]=10-i; for (int i=0; i<=4;i++) { a[i]=a[i+5]; S+=a[i]; }</pre>

20. Нйже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает число A . Укажите такое число x , при вводе которого алгоритм печатает 22935.

Бейсик	Паскаль
<pre> INPUT X K = 1 WHILE X > 0 IF (X MOD 10) MOD 2 = 0 THEN A = A * 10 + X MOD 10 ELSE K = K * 10 B = B * 10 + X MOD 10 END IF X = X \ 10 WEND A = A * K + B PRINT A</pre>	<pre> var x, K, A, B: integer; begin readln(x); K := 1; while x > 0 do begin if (x mod 10) mod 2 = 0 then A:= A*10 + x mod 10 else begin K := K * 10; B:=B*10 + x mod 10 end; x := x div 10 end; A := A * K + B; writeln(A) end.</pre>

Алгоритмический язык	Си
<pre> алг нач цел x, K, A, B ввод x K := 1 нц пока x > 0 если mod(mod(x,10),2)=0 то A := A*10 + mod(x,10) иначе K := K*10 B := B*10 + mod(x,10) все x := div(x,10) кц A := A*K + B вывод A кон </pre>	<pre> #include <stdio.h> void main() { int x, K, A, B; scanf ("%d", &x); K = 1; while (x > 0) { if ((x % 10) % 2 == 0) A = A * 10 + x % 10; else { K *= 10; B = B * 10 + x % 10;} x = x / 10; } A = A * K + B; printf("%d ", A); } </pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

Бейсик	Си
<pre> A = -10: E = 1 WHILE ABS(F1(A)-F2(A))>E A = A+1 WEND NEXT A FUNCTION F1 (X) F1 = 3+2*X END FUNCTION FUNCTION F2 (X) F2 = X-1 END FUNCTION </pre>	<pre> #include<stdio.h> int F1(int x) { return (3+2*x); } int F2(int x) { return (x-1); } void main() { int a,E; a=-10; E=1; while (abs(F1(a)-F2(a))>E) a++; printf("%d", a); } </pre>

Алгоритмический язык	Паскаль
<pre> алг нач цел а, Е а:=-10; Е:=1 нц пока iabs(F1(a)-F2(a))>Е а:=а+1 кц вывод а кон алг цел F1(цел х) знач := 3+2*х кон алг цел F2(цел х) нач знач := х-1 кон </pre>	<pre> var a, E: integer; Function F1(x:integer):integer; begin F1:=3+2*x end; Function F2(x:integer):integer; begin F2:=x-1 end; BEGIN a:=-10; E:=1; while abs(F1(a)-F2(a))>E do a:=a+1; writeln(a) END. </pre>

22. У исполнителя 1Х1Х две команды, которым присвоены номера:

1. прибавь 1,
2. увеличь каждый разряд числа на 1.

Если перед выполнением команды 2 какая-либо цифра числа равна 9, то при выполнении этой команды она не изменяется. Например: при помощи команды 2 число 48 преобразуется в 59, а число 29 в 39 (так как младший разряд нельзя увеличить). Программа для исполнителя 1Х1Х — это последовательность команд. Определите количество программ, которые число 16 преобразуют в число 48?

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_7, x_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$\begin{aligned}
 &\neg(x_1 \equiv x_2) \wedge (x_2 \vee x_3) \wedge (\neg x_2 \vee \neg x_3) = 0 \\
 &\neg(x_2 \equiv x_3) \wedge (x_3 \vee x_4) \vee (\neg x_3 \vee \neg x_4) = 0 \\
 &\dots \\
 &\neg(x_6 \equiv x_7) \wedge (x_7 \vee x_8) \wedge (\neg x_7 \vee \neg x_8) = 0
 \end{aligned}$$

В ответе не нужно перечислять все различные наборы значений $x_1, x_2, \dots, x_7, x_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число N , не превосходящее 10^9 , и выводится максимальная нечётная цифра этого числа. (Гарантируется, что число содержит хотя бы одну нечётную цифру.) Программист торопился и написал программу неправильно.

Алгоритмический язык	
алг	
нач	
цел N, VAL, MAX	
ввод N	
MAX:=9	
нц пока N > 10	
VAL:= mod(N,10)	
если VAL > MAX и mod(VAL,2) = 1 то	
MAX:=VAL	
все	
N:=div(N,10)	
кц	
вывод MAX	
кон	
Паскаль	
var N: longint;	
MAX, VAL: integer;	
begin	
readln(N);	
MAX:=9;	
while N > 10 do	
begin	
VAL:= N mod 10;	
if (VAL>MAX) and (VAL mod 2 = 1)	
then	
MAX:= VAL;	
N:= N div 10	
end;	
writeln(MAX)	
end.	

Бейсик

```
DIM N AS LONG
INPUT N
MAX = 9
WHILE N > 10
    VAL = N MOD 10
    IF VAL > MAX AND VAL MOD 2 = 1 THEN
        MAX=VAL
    END IF
    N = N \ 10
WEND
PRINT MAX
END
```

Си

```
#include <stdio.h>
void main() {
    long N;
    int VAL, MAX;
    scanf("%ld", &N);
    MAX=9;
    while (N>10) {
        VAL = N % 10;
        if (VAL > MAX && VAL % 2 == 1)
            MAX = VAL;
        N /= 10;
    }
    printf("%d", MAX);
}
```

Последовательно выполните следующее.

1. Напишите, что выведет программа при вводе числа 713.
2. Приведите пример такого числа, что, несмотря на ошибки, программа печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;

2) укажите, как исправить ошибку, — приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в данной программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25. Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от 0 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести индекс минимального среди трёхзначных элементов массива, делящихся на 3. Если в исходном массиве нет трёхзначного элемента, кратного 3, то следует вывести сообщение «Не найдено». Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Бейсик
<pre>const N=20; var a: array [1..N] of integer; i,k,min: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>CONST N = 20 DIM A(N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM MIN AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>
Си	Естественный язык
<pre>#include <stdio.h> #define N 20 void main(void){ int a[N]; int i, k, min; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 20 элементов. Объявляем целочисленные переменные I, K и MIN. В цикле от 1 до 20 вводим элементы массива А с 1-го по 20-й. <pre>...</pre>

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого вами языка программирования.

26. Два участника играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(3, 2)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами (x, y) в одну из трёх точек: или в точку с координатами $(x + 1, y)$, или в точку с координатами $(x + 2, y)$, или в точку с координатами $(x, y + 2)$.

Выигрывает игрок, после хода которого расстояние от фишки до точки с координатами $(0, 0)$ не меньше 8 единиц. Кто выигрывает при правильной игре? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

27. На вход программе поступают сведения о заводах, подающих заявку на участие в некотором государственном тендере. В первой строке сообщается количество заявок N , каждая из следующих N строк имеет формат <Название Завода> <Номер Лицензии Завода> <Номер региона>, где <Название Завода> — строка, состоящая не более чем из 50 символов, <Номер Лицензии Завода> — шестизначное число, <Номер региона> — не более чем двузначное натуральное число. <Название Завода> и <Номер Лицензии Завода>, а также <Номер Лицензии Завода> и <Номер региона> разделены одним пробелом.

Пример входных данных:

Ростсельмаш 023398 61

Спринт 342901 77

Рубин 034221 61

Армалит 822145 93

Требуется написать как можно более эффективную программу, которая будет выводить на экран информацию о том, из какого региона было меньше всего заявок (таких регионов может быть несколько). При этом необходимо вывести информацию только по тем регионам, в которых есть завод, подавший заявку. Следует учитывать, что $N \geq 100$.

Пример выходных данных:

регион(ы) с наименьшим числом заявок:

77

93

Вариант № 9

Часть 1

1. Для передачи данных по каналу связи используется 5-битовый код. Сообщение содержит только буквы А, Б и В, которые кодируются следующими кодовыми словами:

А — 10010, Б — 11111, В — 00101.

При передаче возможны помехи. Однако некоторые ошибки можно попытаться исправить. Любые два из этих трёх кодовых слов отличаются друг от друга не менее чем в трёх позициях. Поэтому если при передаче слова произошла ошибка не более чем в одной позиции, то можно сделать обоснованное предположение о том, какая буква передавалась. (Говорят, что «код исправляет одну ошибку».) Например, если получено кодовое слово 10101, считается, что передавалась буква В. (Отличие от кодового слова для В — только в одной позиции, для остальных кодовых слов отличий больше.) Если принятое кодовое слово отличается от кодовых слов для букв А, Б, В более чем в одной позиции, то считается, что произошла ошибка (она обозначается «х»).

Получено сообщение 10011 10111 11100 00111. Декодируйте это сообщение — выберите правильный вариант.

1) АБВВ

2) АБхВ

3) АБхх

4) хххх

2. Дан фрагмент таблицы истинности выражения F :

X	Y	Z	F
1	1	0	0
1	1	1	1
0	1	0	0
0	0	0	1

Каким выражением может быть F ?

- 1) $(\neg Y \vee Z) \rightarrow X$
- 2) $(X \vee 0) \wedge (Y \rightarrow Z)$
- 3) $(X \wedge 1) \vee (\neg Y \vee \neg Z)$
- 4) $(Y \rightarrow X) \wedge (X \rightarrow Z)$

3. Для групповых операций с файлами используются маски имен файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы: символ «?» (вопросительный знак) означает ровно один произвольный символ; символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

Дана группа файлов:

domino.exe	s_found.eps
connect.ex	mount.e
wordicon.exe	takeown.ent
soun_ins.tex	allfont.exe

Определите, сколько по маске $*o?n*.e?*$ будет отобрано файлов.

- 1) 8
- 2) 5
- 3) 3
- 4) 4

4. Сколько из двоичных чисел:

10001010, 00111110, 01100010, 01001110

больше, чем $2_{10}^6 + 2_{10}^3 + 2_{10}^2$?

5. На рисунке 19 изображена схема дорог между населёнными пунктами и обозначены расстояния (в км) между ними. (Населённые пункты обозначены латинскими буквами.) Определите маршрут, по которому можно попасть из пункта A в пункт G , преодолев наименьшее количество километров. В ответе укажите последовательность букв, через которые проходит маршрут.

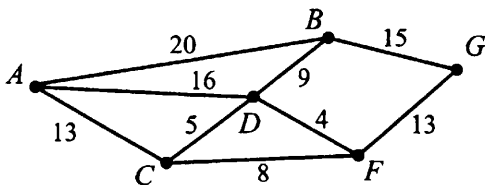


Рис. 19.

6. Дан алгоритм преобразования одного двузначного числа, не содержащего 0, в другое.

1. Возвести в квадрат наибольшую из цифр заданного числа и приписать полученное число в конец исходного.

2. В полученном числе оставить последние две цифры, остальные отбросить.

Укажите наименьшее число, из которого в результате пятикратного выполнения алгоритма будет получено число 36.

7. В ячейке E21 электронной таблицы записана формула. Эту ячейку скопировали в ячейку C12. В результате в ячейке C12 стала записана формула $=2*\$D10+B\$10-C3+3*\$E\6 . Какой вид имеет формула ячейки E21?

1) $= 2 * \$D10 + B\$10 - C3 + 3 * \$E\6

2) $= 2 * \$F19 + D\$10 - E12 + 3 * \$G\6

3) $= 2 * \$D19 + D\$10 - E12 + 3 * \$E\6

4) $= 2 * \$D19 + D\$19 - E12 + 3 * \$E\15

8. Определите, что будет напечатано в результате работы следующего фрагмента программы, записанного ниже на различных языках программирования (переменные k, n, i — целого типа, f — логического типа).

Бейсик
<pre> I = 0: F = 1 : N = 8: K = 3 DO I = I + 1 : N = N - 2*I IF (N > 0) THEN K = K + 1 ELSE F = 0 LOOP UNTIL I = 5 OR NOT(F = 1) PRINT K </pre>
Алгоритмический язык
<pre> i := 0; f := да; n := 8; k := 3 нц i := i + 1; n := n - 2*i если n > 0 то k := k + 1 иначе f := нет все кц_при i = 5 или не f вывод k </pre>

Паскаль
<pre>i := 0; f := true; n := 8; k := 3; repeat i := i + 1; n := n - 2*i; if n > 0 then k := k + 1 else f := false until (i = 5) or (not f); write(k);</pre>
Си
<pre>i = 0; f = true; n = 8; k = 3; do { i++; n -= 2*i; if (n > 0) k++; else f = false; } while (!(i == 5 !f)); printf(" %d",k);</pre>

9. Укажите минимальный объём памяти (в байтах), достаточный для хранения любого растрового изображения размером 16×24 пикселей, если известно, что в изображении используется палитра из 256 цветов. Саму палитру хранить не нужно.

10. Для передачи аварийных сигналов договорились использовать специальные цветные сигнальные флажки, поднимаемые последовательно. Одна последовательность флажков — один сигнал; в каком порядке идут цвета — существенно. Какое количество различных сигналов можно передать при помощи поднятия ровно трёх таких сигнальных флажков, если в запасе имеются флажки шести различных цветов (флажков каждого вида неограниченное количество, цвет флажка в последовательности может повторяться)?

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Паскаль	Си
<pre>function F(n:integer):integer; var t:integer; begin if n=0 then t:=1 else if n mod 2 =0 then t:=F(n div 2) else t:=F(n-1)+n write(t, ' '); F:=t end;</pre>	<pre>int F(int n) { int t; if (n==0) t=1; else if (n % 2 ==0) t=F(n/2); else t=F(n-1)+n; printf("%d ", t); return t; }</pre>
Бейсик	Алгоритмический язык
<pre>FUNCTION F(n) IF n=0 THEN t = 1 ELSE IF n MOD 2 = 0 THEN t = F(n \ 2) ELSE t = F(n - 1) + n END IF END IF PRINT t, F = t END FUNCTION</pre>	<pre>алг цел F(цел n) нач цел t если n=0 то t:=1 иначе если mod(n,2)=0 то t:=F(div(n,2)) иначе t:=F(n-1)+(n+1) все все вывод t, ' ' знач:=t кон</pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(18)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес.

Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 216.126.24.5

Маска: 255.255.192.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
0	5	255	216	126	168	249	165

Пример. Пусть искомый IP-адрес 192.168.128.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

В этом случае правильный ответ будет записан в виде: HBAF.

13. В некоторой библиотеке принята кодировка книг 7 символами. В качестве символов используют 15 букв латинского алфавита и десятичные цифры в любом порядке. Каждый код в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит). Определите объём памяти (в байтах), отводимый этой программой для записи 120 кодов. В ответе запишите только число, слово «байт» писать не нужно.

14. Положение исполнителя Жук, «ползающего» в прямоугольном лабиринте на плоскости (см. рис. 20), характеризуется клеткой, где находится Жук, и направлением, куда смотрит Жук (влево, вправо, вверх, вниз).

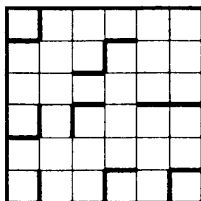


Рис. 20.

Жук может проверить, есть ли перед ним стена (команда *свободно*). Также Жук может выполнять две команды: *вперёд* и *поворот*.

По команде **вперёд** он перемещается на одну клетку в направлении, куда смотрит.

По команде **поворот** Жук поворачивается налево на 90° , оставаясь в той же клетке.

Жук должен выполнить программу:

НАЧАЛО

ЕСЛИ (свободно) вперёд ИНАЧЕ поворот

поворот

ЕСЛИ (свободно) вперёд ИНАЧЕ поворот

поворот

КОНЕЦ

Известно, что изначально Жук смотрит влево. Сколько клеток лабиринта удовлетворяют условию: начиная с них, Жук после выполнения программы по-прежнему смотрит влево?

15. На рисунке 21 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?

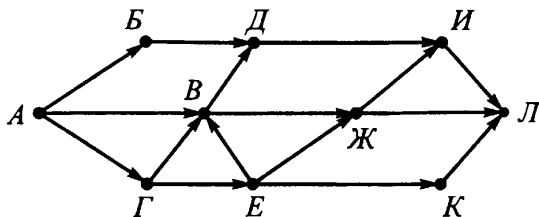


Рис. 21.

16. Сколько значащих нулей содержится в двоичной записи числа, которое можно представить в виде $8^{650} - 4^{810} - 33$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Информация	120
Кодирование	270
Алгоритм	250
Алгоритм Информация	320
Кодирование & (Алгоритм Информация)	90
Алгоритм & (Информация Кодирование)	110

Сколько страниц будет найдено по запросу

Информация | Кодирование?

18. Укажите значения переменных A , B и C , при которых логическое выражение $((A \vee B) \rightarrow B) \vee (\neg C \rightarrow (B \wedge C))$ ложно.

Ответ запишите в виде строки из 3 символов: значений переменных A , B и C (в указанном порядке). Например, строка 001 соответствует тому, что $A = 0$, $B = 0$, $C = 1$.

19. В программе используется одномерный числовой массив A с индексами от 0 до 9. Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> S=0 : K=0 : T=0 A(0)=5 FOR I=1 TO 9 A(I)=I-A(I-1) NEXT I FOR I=0 TO 9 IF A(i)>K THEN K=A(I) END IF IF A(i)<T THEN T=A(I) END IF NEXT I S=K+T </pre>	<pre> S:=0; k:=0; t:=0 A[0]:=5 нц для i от 1 до 9 A[i]:=i-A[i-1] кц нц для i от 0 до 9 если A[i]>k то k:=A[i] все если A[i]<t то t:=A[i] все кц S:=k+t </pre>

Паскаль	Си
<pre> S:=0; k:=0; t:=0; a[i]:=5; for i:=1 to 9 do a[i]:=i-a[i-1]; for i:=0 to 9 do begin if A[i]>k then k:=A[i]; if A[i]<t then t:=A[i] end; S:=k+t; </pre>	<pre> S=0; k=0; t=0; a[i]:=5; for (int i=1; i<=9;i++) a[i]=i-a[i-1]; for (int i=0; i<=9;i++) { if (A[i]>k) k=A[i]; if (A[i]<t) t=A[i]; } S=k+t; </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наибольшее из таких чисел x , при вводе которого алгоритм печатает сначала 3, а потом 11.

Бейсик	Паскаль
<pre> DIM X AS INTEGER DIM A AS INTEGER DIM B AS INTEGER INPUT X A = 0 B = 0 WHILE X > 0 A = A + 1 IF X MOD 2 = 1 THEN B = B + X MOD 10 END IF X = X \ 10 WEND PRINT A PRINT B </pre>	<pre> var x, a, b : integer; begin readln(x); a := 0; b := 0; while x > 0 do begin a := a + 1; if x mod 2 = 1 then b := b + x mod 10; x := x div 10 end; writeln(a); writeln(b); end. </pre>

Алгоритмический язык	Си
<pre> алг нач цел x, a, b ввод x a := 0 b := 0 нц пока x > 0 a := a + 1 если mod(x, 2)=1 то b := b + mod(x, 10) все x := div(x, 10) кц вывод a, nc, b кон </pre>	<pre> #include <stdio.h> void main() { int x, a, b; scanf("%d", &x); a = 0; b = 0; while (x > 0) { a += 1; if x % 2 == 1 b = b + x % 10; x /= 10; } printf("%d \n %d", a, b); } </pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма:

Алгоритмический язык
<pre> алг нач цел a,s,K a:=111111; K:=0 нц для s от a до 222222 если F(mod(s,1000))<3 то если F(mod(s,1000)) = F(div(s,1000)) то K:=K+1 все все кц вывод K кон </pre> алг цел F(цел x) <pre> нач знач =div(div(x,10),10)+mod(div(x,10),10)+mod(x,10) кон </pre>

Бейсик

```

A = 111111 : K = 0
FOR S = A TO 222222
  IF F(S MOD 1000) < 3 THEN
    IF F(S MOD 1000) = F(S \ 1000) THEN K = K + 1
  END IF
NEXT S
PRINT K
FUNCTION F(X)
  F = (X \ 10) \ 10 + (x \ 10) MOD 10 + X MOD 10
END FUNCTION

```

Паскаль

```

var a,s,K: longint;
Function F(x:integer):integer;
begin
  F:=(x div 10) div 10 + (x div 10) mod 10 + x mod 10;
end;
BEGIN
  a:=111111; K:=0;
  for s:=a to 222222 do
    if F(s mod 1000)<3 then
      if F(s mod 1000) = F(s div 1000) then K:=K+1
    writeln(K)
  END.

```

Си

```

#include<stdio.h>
int F(int x) {
  return ((X/10) / 10 + (x/10) % 10 + X % 10);"
}
void main() {
  long a,s,K;
  a=111111; K=0;
  for (s=a; s<=222222; s++)
    if (F(s%1000)<3)
      if (F(s%1000)==F(s/1000)) K+=1;
  printf("%d", K);
}

```

22. У исполнителя IFP15 две команды, которым присвоены номера:

1. прибавь 1,

2. если число простое, то умножь на 5.

Первая из них увеличивает число на экране на 1, вторая, в случае, если число на экране простое, увеличивает его в 5 раз.

Программа для исполнителя IFP15 — это последовательность команд. Определите количество программ, которые число 1 преобразуют в число 56.

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, x_3, x_4, x_5, y_1, y_2, y_3, y_4, y_5$, которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \rightarrow x_2) \wedge (\neg x_1 \rightarrow x_3) \wedge (x_1 \rightarrow x_4) \wedge (\neg x_1 \rightarrow x_5) = 1$$

$$(\neg y_1 \rightarrow y_2) \wedge (y_1 \rightarrow y_3) \wedge (\neg y_1 \rightarrow y_4) \wedge (y_1 \rightarrow y_5) = 1$$

$$(\neg x_1 \vee y_1) \wedge (\neg x_1 \vee y_5) = 1$$

В ответе не нужно перечислять все различные наборы значений $x_1, x_2, x_3, x_4, x_5, y_1, y_2, y_3, y_4, y_5$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры поступает последовательность из пяти неотрицательных целых чисел (некоторые числа могут быть одинаковыми).

Программа должна найти в исходной последовательности минимальное и максимальное чётное число и вывести найденные числа на экран.

Если чётных чисел в последовательности нет, требуется на экран вывести «НО». Известно, что вводимые числа меньше 1000.

Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на четырёх языках программирования.

Паскаль

```
const N=5;
var i, X, MinX, MaxX: integer;
begin
  MinX:=1000; MaxX:=0;
  for i := 1 to n do begin
    readln(X);
    if X mod 2 = 0 then begin
      if X < MinX then MinX:=X
      else
        MaxX:= X
      end
    end;
    if MinX <> 0 then begin
      writeln(MinX); writeln(MaxX)
    end
    else writeln('No')
  end.
```

Си

```
#include <stdio.h>
void main() {
  const int N = 5;
  int i, X, MinX, MaxX;
  MinX=1000; MaxX=0;
  for (i = 1; i <= N; i++) {
    scanf("%d ", &X);
    if (X % 2 == 0) {
      if (X < MinX)
        MinX= X;
      else
        MaxX= X;
    }
  }
  if (MinX <> 0) {
    printf("%d\n %d\n", MinX, MaxX);
  }
  else printf("No")
}
```

Алгоритмический язык

```
алг
нач
  цел N=5, i, X, MinX, MaxX
  MinX:=1000; MaxX:=0
  нц для i от 1 до N
    ввод X
    если mod(X, 2) = 0 то
      если X < MinX то MinX:=X
    иначе
      MaxX:=X;
    все
  все
кц
если MinX <> 0 то
  вывод MinX, нс, MaxX
иначе вывод No
все
кон
```

Бейсик

```
CONST N = 5
MinX = 1000 : MaxX = 0
FOR I = 1 TO n
INPUT X
  IF X mod 2 = 0 THEN
    IF x < MinX THEN
      MinX = X
    ELSE
      MaxX = X
    END IF
  END IF
NEXT I
IF MinX <> 0 THEN
  PRINT MinX
  PRINT MaxX
ELSE
  PRINT "No"
END IF
END
```

Последовательно выполните следующее:

1. Напишите, что выведет эта программа при вводе последовательности чисел 3 2 8 7 4.

2. Приведите пример такой последовательности, содержащей хотя бы одно чётное число, что, несмотря на ошибки, программа печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы.

(Это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы.)

25. Дан массив из 30 элементов. Каждый элемент может принимать целочисленные значения от 0 до 100. Опишите на русском языке или на одном из языков программирования алгоритм, который позволит найти произведение двузначных чётных элементов массива, которые начинаются с цифры 5, 6 или 7. Гарантируется, что в исходном массиве есть хотя бы один такой элемент.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них. Исходные данные всегда подобраны так, что результат произведения не выходит за пределы объявленных типов данных.

Паскаль	Бейсик
<pre>const N=30; var a: array [0..N-1] of integer; i,k: integer; prod: longint; begin for i:=0 to N-1 do readln(a[i]); ... end.</pre>	<pre>CONST N = 30 DIM A(N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM PROD AS LONG FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>

Си	Естественный язык
<pre>#include <stdio.h> #define N 30 void main(void){ int a[N]; int i, k; long prod; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 30 элементов. Объявляем целочисленные переменные I, K и PROD. В цикле от 1 до 30 вводим элементы массива А с 1-го по 30-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого вами языка программирования.

26. Два участника играют в игру «Три кучки». В их распоряжении три кучки камней. Каждым ходом игрок может взять от 1 до 3 камней, но только из одной из трёх кучек. Проигрывает тот, кто взял последний камень. Укажите, у какого из игроков есть выигрышная стратегия, и опишите её, если известно, что изначально в первой кучке было 3 камня, во второй тоже 3, а в третьей — 2.

Примечание. Оба игрока стремятся выиграть и поэтому не делают заведомо проигрышные ходы. Говорят, что у игрока есть выигрышная стратегия, если он может играть так, чтобы победить при любых действиях оппонента.

27. Региональный этап олимпиады по экономике проводился для учеников 9–11-х классов, участвующих в общем конкурсе. Каждый участник олимпиады мог набрать от 0 до 50 баллов. Для определения призёров сначала отбираются 45% (с округлением в меньшую сторону) участников, показавших лучшие результаты.

По положению, в случае, если у последнего участника, входящего в 45%, оказывается такое же количество баллов, как и у следующих за ним в итоговой таблице, решение по данному участнику и всем участни-

кам, имеющим с ним равное количество баллов, определяется следующим образом:

— все участники признаются призёрами, если набранные ими баллы больше половины максимально возможных;

— все участники не признаются призёрами, если набранные ими баллы не превышают половины максимально возможных.

Напишите эффективную по времени работы и по используемой памяти программу, которая по результатам олимпиады будет определять, какой минимальный балл нужно было набрать, чтобы стать призёром олимпиады. На вход программе сначала подаётся число участников олимпиады N . В каждой из следующих N строк находится результат одного из участников олимпиады в следующем формате:

<Фамилия> <Имя> <Класс> <Баллы> ,

где <Фамилия> — строка, состоящая не более чем из 20 символов;

<Имя> — строка, состоящая не более чем из 15 символов;

<Класс> — число от 9 до 11;

<Баллы> — целое число от 0 до 60 набранных участником баллов.

<Фамилия> и <Имя>, <Имя> и <класс>, а также <класс> и <баллы> разделены одним пробелом.

Пример входной строки:

Иванов Пётр 10 17

Программа должна выводить минимальный балл призёра. Гарантируется, что хотя бы одного призёра по указанным правилам определить можно.

Вариант № 10

Часть 1

1. Для передачи данных по каналу связи используется 5-битовый код. Сообщение содержит только буквы А, Б и В, которые кодируются следующими кодовыми словами:

А — 10110, Б — 11101, В — 00000.

При передаче возможны помехи. Однако некоторые ошибки можно попытаться исправить. Любые два из этих трёх кодовых слов отличаются друг от друга не менее чем в трёх позициях. Поэтому если при передаче слова произошла ошибка не более чем в одной позиции, то можно

сделать обоснованное предположение о том, какая буква передавалась. (Говорят, что «код исправляет одну ошибку».) Например, если получено кодовое слово 10000, считается, что передавалась буква В. (Отличие от кодового слова для В — только в одной позиции, для остальных кодовых слов отличий больше.) Если принятое кодовое слово отличается от кодовых слов для букв А, Б, В более чем в одной позиции, то считается, что произошла ошибка (она обозначается «х»).

Получено сообщение 10011 11101 11000 10100. Декодируйте это сообщение — выберите правильный вариант.

- 1) xxxx 2) АБхВ 3) хБхА 4) АБВВ

2. Дан фрагмент таблицы истинности выражения F :

X	Y	Z	F
0	1	1	0
1	0	1	1
1	1	0	0
0	0	0	0

Каким выражением может быть F ?

- 1) $\neg Y \vee Z \vee X$ 2) $(X \vee 0) \wedge (Y \rightarrow Z)$
 3) $(X \wedge 1) \vee (\neg Y \vee \neg Z)$ 4) $(Y \rightarrow X) \wedge (X \rightarrow Z)$

3. Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы: символ «?» (вопросительный знак) означает ровно один произвольный символ; символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

Дана группа файлов:

assets.accdt	answers.vct
_base.vct	Bashful.ico
Tasks.cab	dairydash.ico
Base.c	facsimile.ico

Определите, сколько по маске ***as*.?c*** будет отобрано файлов.

- 1) 8 2) 5 3) 3 4) 4

4. Сколько из двоичных чисел:

10011110, 10101000, 10100000, 10100011

больше, чем $2_{10}^7 + 2_{10}^5 + 2_{10}^2$?

5. На рисунке 22 изображена схема дорог между населёнными пунктами и обозначены расстояния (в км) между ними. (Населённые пункты обозначены латинскими буквами.) Определите маршрут, по которому можно попасть из пункта A в пункт G , преодолев наименьшее количество километров. В ответе укажите последовательность букв, через которые проходит маршрут.

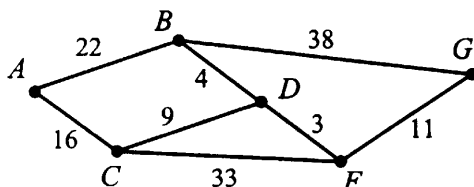


Рис. 22.

6. Четырёхзначное число, состоящее из цифр от 1 до 5, составлено по следующему правилу: на первом месте стоит чётная цифра, на последнем — нечётная, на втором — цифра, равная среднему арифметическому цифр, стоящих на 1-м и на 3-м месте, на третьем — такая цифра, что может быть выполнено условие для цифры, стоящей на втором месте. Укажите наибольшее число, которое может быть составлено по этому правилу.

7. В ячейке C20 электронной таблицы записана формула. Эту ячейку скопировали в ячейку D11. В результате в ячейке D11 стала записана формула $=E7-2*E\$10-3*\$E3+C\$13$. Какой вид имеет формула ячейки C20?

1) $= E7 - 2 * E\$10 - 3 * \$E3 + C\$13$

2) $= D16 - 2 * D\$10 - 3 * \$E12 + B\$13$

3) $= D7 - 2 * D\$1 - 3 * \$F3 + B\$13$

4) $= D7 - 2 * E\$1 - 3 * \$E12 + C\$4$

8. Определите, что будет напечатано в результате работы следующего фрагмента программы, записанного ниже на различных языках программирования (переменные k, n, i — целого типа, f — логического типа).

Бейсик
<pre>I = 8 : F = 1 K = 15 DO I = K MOD I N = K \ I IF N < I THEN K = K + 1 ELSE F = 0 LOOP UNTIL I = 1 OR NOT(F = 1) PRINT K</pre>
Алгоритмический язык
<pre>i := 8; k := 15 f := да нц i := mod(k, i) n := div(k, i) если n < i то k := k + 1 иначе f := нет все кц_при i = 1 или не f вывод k</pre>
Паскаль
<pre>i := 8; k := 15; f := true; repeat i := k mod i; n := k div i; if n < i then k := k + 1 else f := false; until (i = 1) or (not f); write(k);</pre>
Си
<pre>i = 8; f = true; k = 15; do { i = k % i; n = k / i; if (n < i) k++; else f = false; } while (!(i == 1 !f)); printf("%d ", k);</pre>

9. Для хранения растрового изображения размером 12×25 пикселя отвели 225 байтов памяти. Каково максимально возможное число цветов в палитре изображения?

10. Для передачи аварийных сигналов договорились использовать специальные цветные сигнальные флажки, поднимаемые последовательно. Одна последовательность флажков — один сигнал; в каком порядке идут цвета — существенно. Какое количество различных сигналов можно передать при помощи поднятия ровно пяти таких сигнальных флажков, если в запасе имеются флажки четырёх различных цветов (флажков каждого вида неограниченное количество, цвет флажка в последовательности может повторяться)?

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n=0 THEN t = 1 ELSE IF n MOD 3 = 0 THEN t = F(n \ 3) ELSE t = F(n - 1) + 2*n END IF END IF PRINT t, F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=0 то t:=1 иначе если mod(n,3)=0 то t:=F(div(n,3)) иначе t:=F(n-1)+2*(n+1) все все вывод t, ' ' знач:=t кон </pre>
Паскаль	Си
<pre> function F(n:integer):integer; var t:integer; begin if n=0 then t:=1 else if n mod 3 =0 then t:=F(n div 3) else t:=F(n-1)+2*n write(t, ' '); F:=t end; </pre>	<pre> int F(int n) { int t; if (n==0) t=1; else if (n%3==0) t=F(n/3); else t=F(n-1)+2*n; printf("%d ", t); return t; } </pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(21)?

12. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 192.124.16.28

Маска: 255.255.240.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы. Точки писать не нужно.

A	B	C	D	E	F	G	H
255	240	16	216	124	192	28	0

Пример. Пусть искомый IP-адрес 168.53.224.0, и дана таблица

A	B	C	D	E	F	G	H
128	168	255	15	53	0	17	224

В этом случае правильный ответ будет записан в виде: BEHF.

13. В некоторой библиотеке принята кодировка книг 9 символами. В качестве символов используют 13 букв латинского алфавита и десятичные цифры в любом порядке. Каждый код в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит). Определите объём памяти (в байтах), отводимый этой программой для записи 50 кодов. В ответе запишите только число, слово «байт» писать не нужно.

14. Положение исполнителя Жук, «ползающего» в прямоугольном лабиринте на плоскости (см. рис. 23), характеризуется клеткой, где находится Жук, и направлением, куда смотрит Жук (влево, вправо, вверх, вниз).

Жук может проверить, есть ли перед ним стена (команда свободно). Также Жук может выполнять две команды: **вперёд** и **поворот**. По команде **вперёд** он перемещается на одну клетку в направлении, куда смотрит. По команде **поворот** Жук поворачивается налево на 90°, оставаясь в той же клетке.

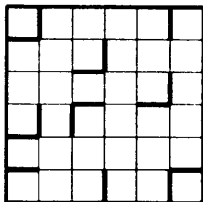


Рис. 23.

Жук должен выполнить программу:

НАЧАЛО

ЕСЛИ (свободно) вперёд ИНАЧЕ поворот

поворот

ЕСЛИ (свободно) вперёд ИНАЧЕ поворот

поворот

КОНЕЦ

Сколько клеток лабиринта удовлетворяют условию: начиная с некоторого положения в этой клетке, Жук после выполнения программы останется в одной из четырёх угловых клеток лабиринта?

15. На рисунке 24 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?

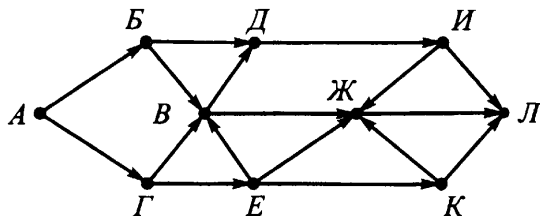


Рис. 24.

16. Сколько единиц содержится в двоичной записи числа, которое можно представить в виде $16^{500} - 4^{760} - 36$?

17. На языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ $|$, а для логической операции «И» — символ $\&$.

Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов некоторого сегмента сети. Вот её фрагмент:

Ключевое слово	Количество страниц (в сотнях тысяч)
Валюта	320
Кредит	190
Рынок	380
Рынок Валюта	600
Кредит & (Рынок Валюта)	130
Валюта Кредит	410

Сколько страниц будет найдено по запросу
Рынок & (Валюта | Кредит)?

18. Укажите значения переменных A , B и C , при которых логическое выражение $(A \rightarrow (B \vee \neg C)) \rightarrow (B \wedge (B \rightarrow (A \wedge \neg B)))$ истинно.

Ответ запишите в виде строки из 3 символов: значений переменных A , B и C (в указанном порядке). Например, строка 001 соответствует тому, что $A = 0$, $B = 0$, $C = 1$.

19. В программе используется одномерный числовой массив A с индексами от 0 до 9. Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Паскаль	Си
<pre> S:=0; k:=0; t:=0; a[i]:=-5; for i:=1 to 9 do a[i]:=2*i-a[i-1]; for i:=0 to 8 do begin if A[i]+A[i+1]>k then k:=A[i]+A[i+1]; if A[i]*A[i+1]<t then t:=A[i]*A[i+1]] end; S:=k+t; </pre>	<pre> S=0; k=0; t=0; a[i]:=-5; for (int i=1; i<=9;i++) a[i]=2*i-a[i-1]; for (int i=0; i<9;i++) { if (A[i]+A[i+1]>k) k=A[i]+A[i+1]; if (A[i]*A[i+1]<t) t=A[i]*A[i+1]; } S=k+t; </pre>

Бейсик	Алгоритмический язык
<pre> S=0 K=0 T=0 A(0)=-5 FOR I=1 TO 9 A(I)=2*I-A(I-1) NEXT I FOR I=0 TO 8 IF A(i)+A(I+1)>K THEN K=A(i)+A(I+1) END IF IF A(i)*A(I+1)<T THEN T=A(i)*A(I+1) END IF NEXT I S=K+T </pre>	<pre> S:=0 k:=0 t:=0 A[0]:=-5 нц для i от 1 до 9 A[i]:=2*i-A[i-1] кц нц для i от 0 до 8 если A[i]+A[i+1]>k то k:=A[i]+A[i+1] все если A[i]*A[i+1]<t то t:=A[i]*A[i+1] все кц S:=k+t </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает число S . Укажите такое число x , при вводе которого алгоритм печатает число 31972.

Бейсик	Паскаль
<pre> INPUT X S = 0 WHILE X > 0 S = S * 10 + X MOD 10 X = X \ 10 WEND PRINT S </pre>	<pre> var x, S: integer; begin readln(x); S := 0; while x > 0 do begin S := S * 10 + x mod 10; x := x div 10 end; writeln(S); end. </pre>

Алгоритмический язык	Си
алг нач цел x, S ввод x S := 0 нц пока x > 0 S := S * 10 + mod(x, 10) x := div(x, 10) кц вывод S кон	<pre>#include <stdio.h> void main() { int x, S; scanf("%d", &x); S = 0; while (x > 0) { S = S * 10 + x % 10; x /= 10; } printf("%d", S); }</pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма:

Алгоритмический язык
алг нач цел a, s, L a:=222222 L:=F(mod(a,1000)) нц для s от a до 333333 если F(mod(s,1000)) = F(div(s,1000)) то если L > F(mod(s,1000)) то L:=F(mod(s,1000)) все все кц вывод L кон алг цел F(цел x) нач знач:=div(div(x,10),10)+mod(div(x,10),10)+mod(x,10) кон

Паскаль

```
var a,s,L: longint;
Function F(x:integer):integer;
begin
    F=(x div 10) div 10 + (x div 10) mod 10 + x mod 10;
end;

BEGIN
    a:=222222;
    L:=F(a mod 1000);
    for s:=a to 333333 do
        if F(s mod 1000) = F(s div 1000) then
            if L > F(s mod 1000) then
                L:=F(s mod 1000);
            writeln(L)
        end if
    end for
END.
```

Бейсик

```
A = 222222
L = F(A MOD 1000)
FOR S = A TO 333333
    IF F(S MOD 1000) = F(S \ 1000) THEN
        IF L>F(S MOD 1000) THEN
            L = F(S MOD 1000)
        END IF
    END IF
NEXT S
PRINT L

FUNCTION F(X)
    F = (X\10) \ 10 + (x\10) MOD 10 + X MOD 10
END FUNCTION
```

Си

```
#include<stdio.h>
int F(int x) {
    return ((X/10) / 10 + (x/10) % 10 + x % 10);
}
void main() {
long a,s,L;
    a=222222;
    L=F(a%1000);
    for (s=a; s<=333333; s++)
        if (F(s%1000)==F(s/1000))
            if (L > F(s%1000))
                L= F(s%1000);
    printf("%d", L);
}
```

22. У исполнителя IFP13 две команды, которым присвоены номера:

1. прибавь 1,

2. если число простое, то умножь на 3.

Первая из них увеличивает число на экране на 1, вторая — в случае, если число на экране простое, увеличивает его в 3 раза. Программа для исполнителя IFP13 — это последовательность команд.

Определите количество программ, которые число 1 преобразуют в число 42.

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, x_3, x_4, x_5, y_1, y_2, y_3, y_4, y_5$, которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \rightarrow \neg x_2) \wedge (x_1 \rightarrow \neg x_3) \wedge (x_1 \rightarrow \neg x_4) \wedge (x_1 \rightarrow \neg x_5) = 1$$

$$(\neg y_1 \rightarrow y_2) \wedge (y_2 \rightarrow \neg y_3) \wedge (\neg y_3 \rightarrow y_4) \wedge (y_4 \rightarrow \neg y_5) = 1$$

$$(\neg x_1 \vee y_1) \wedge x_1 = 1$$

В ответе **не нужно** перечислять все различные наборы значений $x_1, x_2, x_3, x_4, x_5, y_1, y_2, y_3, y_4, y_5$, при которых выполнена данная система равенств. В качестве ответа нужно указать количество таких наборов.

Часть 2

24. Требовалось написать программу, которая получает с клавиатуры координаты точки на плоскости (x, y — действительные числа) и определяет принадлежность точки заштрихованной области, включая её границы (см. рис. 25). Программист написал программу неправильно.

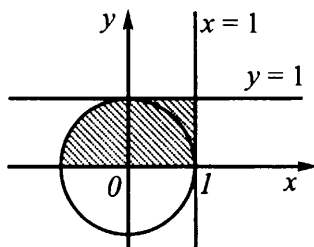


Рис. 25.

Бейсик

```

INPUT x,y
IF y <= 1 THEN
  IF x <= 1 THEN
    IF x*x+y*y <= 1 THEN
      PRINT "ПРИНАДЛЕЖИТ"
    ELSE
      PRINT "НЕ ПРИНАДЛЕЖИТ"
    END IF
  END IF
END IF
END

```

Паскаль

```

var x,y : real;
begin
  readln(x,y);
  if y<=1 then
    if x<=1 then
      if x*x+y*y<=1 then
        writeln('ПРИНАДЛЕЖИТ')
      else
        writeln('НЕ ПРИНАДЛЕЖИТ')
      end if
    end if
  end if
end.

```

Си
<pre>#include<stdio.h> void main() { float x, y; scanf(" %f %f", &x, &y); if (y<=1) if (x<=1) if (x*x+y*y<=1) printf("ПРИНАДЛЕЖИТ"); else printf("НЕ ПРИНАДЛЕЖИТ"); } }</pre>

Выполните следующее:

1. Приведите пример таких чисел x, y , при которых программа неверно решает поставленную задачу.

2. Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

25. Дан массив из 30 элементов. Каждый элемент может принимать целочисленные значения от 0 до 100. Опишите на русском языке или на одном из языков программирования алгоритм, который позволит найти сумму двузначных элементов массива, которые не оканчиваются на 3. Гарантируется, что в исходном массиве есть хотя бы один такой элемент. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них. Исходные данные всегда подобраны так, что результат суммы не выходит за пределы объявленных типов данных.

Паскаль	Бейсик
<pre>const N=30; var a: array [0..N-1] of integer; i,k: integer; sum: longint; begin for i:=0 to N-1 do readln(a[i]); ... end.</pre>	<pre>N = 30 DIM A(N) AS INTEGER DIM I AS INTEGER DIM K AS INTEGER DIM SUM AS LONG FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>

Си	Естественный язык
<pre>#include <stdio.h> #define N 30 void main(void){ int a[N]; int i, k; long sum; for (i=0; i<N; i++) scanf("%d", &a[i]); ...}</pre>	Объявляем массив А из 30 элементов. Объявляем целочисленные переменные I, K и SUM. В цикле от 1 до 30 вводим элементы массива А с 1-го по 30-й. ...

В качестве ответа необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае Вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учётом синтаксиса и особенностей используемого Вами языка программирования.

26. Два игрока играют в «Верёвку». Игроки ходят по очереди. В начале игры верёвка имеет длину 18 см. Ход состоит в том, что игрок отрезает от верёвки кусок длиной 4 или 5 см. Выигрывает тот игрок, на чьём ходе закончится верёвка (последний выигрышный ход может быть < 4). Кто выиграет при безошибочной игре двух игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Ответ обоснуйте.

27. В 64-квартирном доме проводится проверка долгов жильцов по оплате коммунальных услуг. Для формирования сообщений о накопившемся долге выбираются номера квартир, долг за которые превышает 80% от максимального долга по всем квартирам. Если долги у всех одинаковые, то выбираются первые 60% квартир-должников, начиная с минимального номера (округлять следует в меньшую сторону, например, при шести должниках будут выбраны первые 3 квартиры-должника).

Напишите эффективную по времени работы и по используемой памяти программу, которая выбирает номера необходимых квартир.

На вход программы сначала подаётся число квартир-должников N . В каждой из следующих N строк находятся сведения о долге одной из квартир в формате: <Фамилия> <Имя> <квартира> <долг>, где <Фамилия> — строка, состоящая не более чем из 20 символов,

<Имя> — строка, состоящая не более чем из 15 символов,

<квартира> — целое положительное число от 1 до 64,

<долг> — положительное вещественное число.

<Фамилия> и <Имя>, <Имя> и <квартира>, <квартира> и <долг> разделены одним пробелом.

Пример входной строки:

Иванов Иван 1 107.39

Программа должна выводить номера квартир-должников, подходящих по условию. Гарантируется, что максимальный долг не превышает 3000 рублей и каждая квартира во вводимых данных присутствует только один раз.

Вариант № 11

Часть 1

1. Для 5 букв русского алфавита заданы их двоичные коды (для некоторых букв коды состоят из двух бит, для некоторых — из трёх). Эти коды представлены в таблице

А	Б	К	Л	О
01	110	11	001	10

Из четырёх полученных сообщений в этой кодировке только одно прошло без ошибки и может быть корректно декодировано. Найдите его.

1) 10011001011100

2) 10110001011110

3) 10000101011110

4) 10111001010010

2. Символом F обозначено логическое выражение от трёх аргументов A , B , C : $(A \rightarrow \neg B) \vee C$. Какой из фрагментов таблиц истинности, приведённых ниже, соответствует данному выражению $F(A, B, C)$?

1)

A	B	C	F
1	1	1	1
0	0	1	1
1	0	0	1

2)

A	B	C	F
0	0	0	0
0	0	1	1
1	1	0	0

3)

A	B	C	F
0	1	0	0
0	0	1	1
1	0	0	1

4)

A	B	C	F
0	1	1	1
0	0	1	0
1	1	1	1

3. Ниже представлен фрагмент таблицы результатов выполнения трёх попыток прыжков в длину с места учащимися 11-го класса.

Ф. И. О.	Пол	Попытка		
		1-я	2-я	3-я
Акимов В. М.	муж	230	242	208
Большова С. Ю.	жен	185	175	170
Верёвкин В. К.	муж	235	232	204
Гнеденко Ф. Б.	жен	195	192	180
Гончаров Л. М.	муж	210	205	210
Денисенко Л. И.	жен	180	198	186
Егоров В. И.	муж	205	209	220

Оценка выставляется по самой успешной из трёх попыток согласно таблицам:

	Оценка в баллах				
	1	2	3	4	5
Мальчики	< 190	190 – 205	206 – 220	221 – 240	> 240
Девочки	< 160	160 – 170	171 – 190	191 – 210	> 210

Определите, какое количество учащихся получили оценку 4 или 5.

- 1) 5 2) 6 3) 3 4) 4

4. Неотрицательные целые числа кодируются своим представлением в двоичной системе счисления. Укажите наименьшее количество бит, которое необходимо для записи числа 245.

5. Между населёнными пунктами *A*, *B*, *C*, *D*, *E* и *F* построены дороги, протяжённость которых приведена в таблице. (Отсутствие числа в таблице означает, что прямой дороги между соответствующими пунктами нет.)

	A	B	C	D	E	F
A	-	4	-	-	-	-
B	4	-	2	4	5	-
C	-	2	-	-	2	-
D	-	4	-	-	2	-
E	-	5	2	2	-	5
F	-	-	-	-	5	-

Определите длину кратчайшего пути между пунктами *A* и *F* (при условии, что передвигаться можно только по построенным дорогам).

6. У исполнителя *Вычислитель* имеется три команды:

1) **умножь на 2**,

2) **умножь на 5**,

3) **прибавь 1**.

Выполняя первую из них, *Вычислитель* удваивает число на экране, выполняя вторую, увеличивает число на экране в 5 раз, а выполняя третью, увеличивает число на 1.

Запишите порядок команд в программе преобразования числа 5 в число 105, содержащий не более 4 команд, указывая лишь номера команд.

Например, 2131 — это программа:

умножь на 5, умножь на 2, прибавь 1, умножь на 2,
которая преобразует число 2 в число 42.

7. Дан фрагмент электронной таблицы:

	A	B
1	2	= МАКС(A1 : A4)
2	3	= МАКС (A1 : A3)
3	4	= СРЗНАЧ (A3; A4) – 4
4		= A1 * 4 – A3

Какое число должно быть записано в ячейке A3, чтобы построенная после выполнения вычислений диаграмма по значениям диапазона ячеек B1:B4 соответствовала рисунку 26?

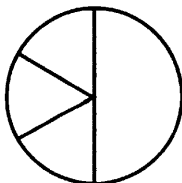


Рис. 26.

8. Определите значение переменной s после выполнения фрагмента программы, записанного ниже на различных языках программирования (переменные s, m, d — целого типа).

Бейсик	Алгоритмический язык
<pre> S = 0 M = 1472 WHILE M > 10 D = M MOD 10 S = S + D M = M \ 10 WEND S = S + M </pre>	<pre> s := 0 m := 1472 нц пока m > 10 d := mod(m, 10) s := s + d m := div(m, 10) кц s := s + m </pre>
Паскаль	Си
<pre> s := 0; m := 1472; while m > 10 do begin d := m mod 10; s := s + d; m := m div 10 end; s := s + m; </pre>	<pre> s = 0; m = 1472; while (m > 10) { d = m % 10; s += d; m /= 10; } s += m; </pre>

9. Сервер провайдера подключён к Интернету по высокоскоростному каналу 10 Мбит в секунду. Пользователь подключён к Интернету через сервер данного провайдера, причём скорость канала между провайдером и пользователем составляет 320 Кбит в секунду. Пользователю требуется скачать из Интернета файл размером 5 Мб. Сервер провайдера начинает трансляцию данных не раньше, чем им получены первые 1280 Кб этих данных.

Каков минимально возможный промежуток времени (в секундах) с момента начала скачивания провайдером данных до полного их получения пользователем?

10. На одном предприятии работает 230 людей. Для каждого рабочего необходимо завести карточку, номер которой состоял бы из цифр 6, 4, 2. При этом все номера карт должны иметь одну длину и быть различными. Определите минимальную длину номера карты.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n = 1 THEN t = 1 ELSE IF n = 0 THEN t = 1 ELSE t = F(n - 1) + F(n - 2) END IF END IF PRINT t, F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=1 то t:=1 иначе если n=0 то t:=1 иначе t:=F(n-1)+F(n-1) все все вывод t, ' ' знач:=t кон </pre>
Паскаль	Си
<pre> function F(n:integer):integer; var t:integer; begin if n=1 then t:=1 else if n=0 then t:=1 else t:=F(n-1)+F(n-2) end end write(t, ' '); F:=t end; </pre>	<pre> int F(int n) { int t; if (n==1) t=1; else if (n==0) t=1; else t=F(n-1)+F(n-2); printf("%d ", t); return t; } </pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(6)?

12. Составьте допустимый IP-адрес по следующим фрагментам:

А	Б	В	Г	Д	Е	Ж
254.	304	16.45	23	236.2	256	47

В ответе укажите последовательность букв, обозначающих фрагменты, в порядке их вхождения в IP-адрес. (Буквы не должны повторяться.)

13. В некоторой стране автомобильный номер длиной 6 символов составляют из заглавных букв (всего используется 22 различные буквы) и десятичных цифр, записанных в любом порядке. Каждый такой номер в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование: каждый символ кодируется одинаковым и минимально возможным количеством бит). Определите объём памяти (в байтах), отводимый этой программой для записи 45 номеров. В ответе запишите только число, слово «байт» писать не нужно.

14. Система команд исполнителя *РОБОТ*, «живущего» в прямоугольном лабиринте на клетчатой плоскости, включает 4 команды-приказа и 4 команды — проверки условия.

Команды-приказы:

вверх, вниз, влево и вправо.

При выполнении любой из этих команд *РОБОТ* перемещается на одну клетку соответственно: вверх, вниз, влево, вправо.

Если *РОБОТ* начнёт движение в сторону находящейся рядом с ним стены, то он разрушится, и программа прервётся.

Другие четыре команды проверяют истинность *условия* отсутствия стены у каждой стороны той клетки, где находится *РОБОТ*:

сверху свободно, снизу свободно, слева свободно, справа свободно.

Цикл

ПОКА <условие>

последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно, иначе происходит переход на выполнение следующей команды.

В конструкции

ЕСЛИ <условие>

ТО команда1

ИНАЧЕ команда2

КОНЕЦ ЕСЛИ

выполняется команда1 (если условие истинно) или команда2 (если условие ложно).

Сколько клеток лабиринта соответствуют требованию, что, начав в ней движение и выполнив предложенную программу, *РОБОТ* уцелеет и остановится в закрашенной клетке F2 (см. рис. 27)?

НАЧАЛО

ПОКА <справа свободно ИЛИ сверху свободно>

ЕСЛИ <справа свободно>

ТО вправо

ИНАЧЕ вверх

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

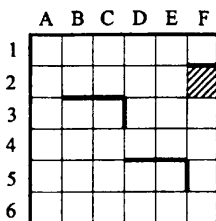


Рис. 27.

15. На рисунке 28 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

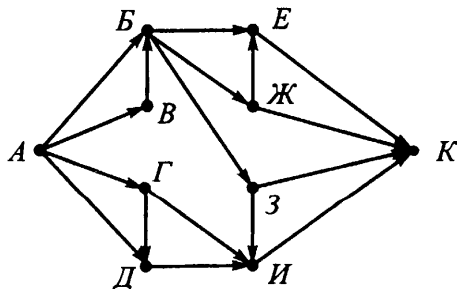


Рис. 28.

16. Сколько цифр в двоичной записи десятичного числа, которое можно представить в виде $2^{530} - 2^{16} - 4^{100}$?

17. В языке запросов поискового сервера для обозначения логической операции «ИЛИ» в запросе используется символ $|$, а для логической операции «И» — символ $\&$. В таблице приведены запросы и количество страниц, найденных поисковым сервером по этим запросам в некотором сегменте Интернета:

Запрос	Найдено страниц (в тысячах)
олимпиада $\&$ информатика	864
(олимпиада $ $ задачи) $\&$ информатика	1020
задачи $\&$ информатика	731

Какое количество страниц (в тысячах) будет найдено по запросу **олимпиада $\&$ задачи $\&$ информатика**?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

18. Укажите количество натуральных значений X , при которых истинно логическое выражение

$$(X^3 < 27) \wedge ((X > 1) \rightarrow (X^2 - 4 < 0)).$$

19. В программе используется двумерный числовой массив A размером 5×5 , все элементы которого равны 1. Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> S = 0 FOR I = 0 TO 4 FOR J = 0 TO 4 IF I = J THEN S = S + A(I, I) ELSE S = S + i + j END IF NEXT J NEXT I </pre>	<pre> S:=0 нц для i от 0 до 4 нц для j от 0 до 4 если i=j то S:=S+A[i,i] иначе S:=S+i+j все кц кц </pre>

Си	Паскаль
<pre> S = 0 for (i=0; i<=4; i++) for (j=0; j<=4; j++) if (i==j) S+=A[i][i]; else S+=(i+j); </pre>	<pre> S:=0 for i:=0 to 4 do for j:=0 to 4 do if i=j then S:=S+A[i,i] else S:=S+i+j; </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа A и B . Укажите наибольшее из таких чисел x , при вводе которых алгоритм печатает сначала 4, а потом 14.

Бейсик	Паскаль
<pre> INPUT X A = 0 : B = 0 WHILE X > 0 A = A + 1 IF B < 2*(X MOD 10) THEN B = 2*(X MOD 10) END IF X = X \ 10 WEND PRINT A PRINT B </pre>	<pre> var x, A, B: integer; begin readln(x); A := 0; B := 0; while x > 0 do begin A := A + 1; if B < 2*(x mod 10) then B := 2*(x mod 10); x := x div 10 end; writeln(A); writeln(B) end. </pre>
Алгоритмический язык	Си
<pre> алг нач цел x, A, B ввод x A := 0; B := 0 нц пока x > 0 A := A + 1 если B < 2*mod(x, 10) то B := 2*mod(x, 10) все x := div(x, 10) кц вывод A, нс, B кон </pre>	<pre> #include<stdio.h> void main() { int x, A, B; scanf ("%d", &x); A = 0; B = 0; while (x > 0) { A = A + 1; if (B < 2 * (x % 10)) B = 2 * (x % 10); x = x / 10; } printf("%d\n%d", A, B); } </pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

Бейсик	Паскаль
<pre> B = 7: L = 0 FOR S = 0 TO B IF S MOD 2 = 0 THEN L = L + F(S) END IF NEXT S PRINT L FUNCTION F (X) IF X = 0 THEN F = 1 ELSE F = X * F(X-1) END IF END FUNCTION </pre>	<pre> var b,s,L: integer; Function F(x:integer):integer; begin if x = 0 then F:=1 else F:=x * F(x-1); end; BEGIN b:=7; L:=0; for s:=0 to b do if s mod 2 = 0 then L:=L+F(s) writeln(L) END. </pre>
Алгоритмический язык	Си
<pre> алг нач цел b,s,L b:=7; L:=0 нц для s от 0 до b если mod(s,2)=0 то L:=L+F(s) все кц вывод L кон алг цел F(цел x) нач если x=0 то знач:=1 иначе знач:=x * F(x-1) все кон </pre>	<pre> #include<stdio.h> int F(int x) { if (x==0) {return 1;} else return (x * F(x-1)); } void main() { int b,s,L; b=7; L=0; for (s=0; s<=b; s++) if (s%2==0) L=L+F(s); printf("%d", L); } </pre>

22. У исполнителя *Уменьшитель* две команды, которым присвоены номера:

1. разделить на 2,
2. вычесть 1.

Первая из них уменьшает число на экране в 2 раза, вторая — уменьшает его на 1. Программа для исполнителя *Уменьшитель* — это последовательность команд.

Определите количество программ, которые число 64 преобразуют в число 14.

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_{10}$, которые удовлетворяют всем перечисленным ниже условиям?

$$\begin{aligned} ((x_1 \equiv x_2) \wedge (\neg x_3 \equiv x_4)) \vee ((\neg x_1 \wedge \neg x_2) \wedge (\neg x_3 \equiv x_4)) &= 0, \\ ((x_3 \equiv x_4) \wedge (\neg x_5 \equiv x_6)) \vee ((\neg x_3 \wedge \neg x_4) \wedge (\neg x_5 \equiv x_6)) &= 0, \\ ((x_5 \equiv x_6) \wedge (\neg x_7 \equiv x_8)) \vee ((\neg x_5 \wedge \neg x_6) \wedge (\neg x_7 \equiv x_8)) &= 0, \\ (((x_7 \equiv x_8) \wedge (\neg x_9 \equiv x_{10})) \vee ((\neg x_7 \wedge \neg x_8) \wedge (\neg x_9 \equiv x_{10}))) &= 0. \end{aligned}$$

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается действительное число x и вычисляется значение функции $y(x)$, заданной условиями:

$$y = \begin{cases} \frac{1}{x}, & \text{если } x > 2, \\ 2x^2 - 3, & \text{если } -1 \leq x \leq 2, \\ |x|, & \text{в остальных случаях.} \end{cases}$$

Программист торопился и написал программу неправильно.

Бейсик
<pre> INPUT x IF x > 2 THEN y = 1/x IF x >= -1 AND x <= 2 THEN y = 2*x*x-3 ELSE y = ABS(x) END IF PRINT y END </pre>

Паскаль
<pre> var x, y: real; begin readln(x); if x>2 then y := 1/x; if (x>=-1) and (x<=2) then y := 2*x*x-3 else y := abs(x); write(y) end.</pre>
Алгоритмический язык
<pre> алг нач вещ x,y ввод x если x>2 то y:=1/x все если x>=-1 и x<=2 то y:=2*x*x-3 иначе y:=iabs(x) все вывод y кон</pre>
Си
<pre> #include <stdio.h> void main() { float x, y; scanf("%f ", &x); if (x>2) y=1/x; if (x>=-1 && x<=2) y=2*x*x-3; else y=abs(x)); printf("%f", y); }</pre>

Последовательно выполните следующее:

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при значениях x , принадлежащих различным промежуткам: $(-\infty; -1)$, $[-1; 2]$, $(2; +\infty)$.

В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.», если программа ведёт себя по-разному для разных значений, принадлежащих данному промежутку. В столбце «Выполняемая функция» запишите функцию, для которой вычисляется значение. В последнем столбце укажите «да» или «нет».

2. Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы. (Это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы.)

Проме- жуток	Усл. 1 $x > 2$	Усл. 2 $-1 \leq x \leq 2$	Выполняемая функция	Промежуток обрабаты- вается верно
$(-\infty; -1)$				
$[-1; 2]$				
$(2; +\infty)$				

25. Дан целочисленный массив из 43 элементов. Элементы массива могут принимать целые значения от 55 до 155. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести максимальное значение среди элементов массива, которые имеют нечётное значение и не делятся на 9. Гарантируется, что в исходном массиве есть хотя бы один такой элемент.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

Бейсик	Алгоритмический язык
CONST N = 43	алг
DIM A(N) AS INTEGER	нач
DIM I AS INTEGER	цел N=43
DIM J AS INTEGER	целтаб a[1:N]
DIM MAX AS INTEGER	цел i, j, max
FOR I = 1 TO N	нц для i от 1 до N
INPUT A(I)	ввод a[i]
NEXT I	кц
...	...
END	кон

Паскаль	Си
<pre>const N=43 var a: array [1..N] of integer; i, j, max: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre>#define N 43 void main(void){ int a[N]; int i, j, max; for (i=0; i<N; i++) scanf("%d", &a[i]); ... }</pre>
Естественный язык	
Объявляем массив A из 43 элементов. Объявляем целочисленные переменные I, J, MAX. В цикле от 1 до 43 вводим элементы массива A с 1-го по 43-й. ...	

В качестве ответа Вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Borland Pascal 7.0) или в виде блок-схемы. В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

26. Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(-2; 1)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами $(x; y)$ в одну из трёх точек: или в точку с координатами $(x + 4; y)$, или в точку с координатами $(x; y + 3)$, или в точку с координатами $(x + 2; y + 2)$. Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами $(0; 0)$ больше 9 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или его партнёр? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

27. На вход программе подаются сведения об акциях, торгуемых на фондовой бирже *NASDAQ*. В первой строке сообщается количество торгу-

емых акций N , каждая из следующих N строк имеет формат <Название Акции> <Тип Операции> <Цена>, где <Название Акции> — строка, состоящая не более чем из 4 символов, <Тип Операции> — строка, состоящая из символа «s» или «b», причём символ «s» означает, что данную акцию можно **продать** по указанной цене, а символ «b» — то, что данную акцию можно **купить** по указанной цене, <Цена> — не более чем трёхзначное целое число. <Название Акции> и <Тип Операции>, а также <Тип Операции> и <Цена> разделены одним пробелом.

Пример входных данных:

MSFT b 26

QQQ s 56

APR b 389

HP s 59

Требуется написать как можно более эффективную программу, которая будет осуществлять стратегию **покупки** всех возможных акций, чья стоимость не превышает приемлемой цены 240, и выводить на экран информацию о количестве и средней стоимости всех приобретённых акций. Следует учитывать, что $N \geq 100\,000$. Также гарантируется, что будет куплена хотя бы 1 акция.

Пример выходных данных:

совершена 1 операция по покупке акций

средняя стоимость 1 приобретённой акции 26 долл.

Вариант № 12

Часть 1

1. Для 5 букв русского алфавита заданы их двоичные коды (для некоторых букв коды состоят из двух бит, для некоторых — из трёх). Эти коды представлены в таблице

А	В	К	Р	Н
01	000	11	001	10

Из четырёх полученных сообщений в этой кодировке только одно прошло без ошибки и может быть корректно декодировано. Найдите его.

1) 1101001010010010

2) 1101001010010011

3) 1101001010000110

4) 1101001010001100

2. Какой из перечисленных ниже фрагментов таблиц истинности соответствует логическому выражению $F = (X \wedge Z) \vee (Y \wedge Z)$?

1)

X	Y	Z	F
1	1	0	0
1	0	0	0
0	0	1	0

2)

X	Y	Z	F
1	1	0	1
1	0	1	1
0	0	1	0

3)

X	Y	Z	F
1	1	1	1
1	0	1	0
0	0	0	0

4)

X	Y	Z	F
0	1	1	1
0	1	0	1
0	0	1	0

3. Ниже представлен фрагмент таблицы результатов выполнения трёх попыток прыжков в длину с места учащимися 11-го класса.

Ф. И. О.	Пол	Попытка		
		1-я	2-я	3-ья
Колосов К. М.	муж	210	242	208
Круглов С. Ш.	муж	185	190	198
Ластенко В. К.	жен	166	168	180
Лимонова З. Т.	жен	195	192	180
Михайлов Е. М.	муж	210	205	210
Николаев Л. Н.	муж	188	198	206
Некрасова В. И.	жен	205	209	220

Оценка выставляется по самой успешной из трёх попыток согласно таблицам:

	Оценка в баллах				
	1	2	3	4	5
Мальчики	< 190	190 – 205	206 – 220	221 – 240	> 240
Девочки	< 160	160 – 170	171 – 190	191 – 210	> 210

Определите, какое количество учащихся получили оценку 3.

1) 5

2) 6

3) 3

4) 4

4. Неотрицательные целые числа кодируются своим представлением в двоичной системе счисления. Укажите наименьшее количество байтов, которое необходимо для записи числа 284.

5. Между населёнными пунктами A, B, C, D, E и F построены дороги, протяжённость которых приведена в таблице. (Отсутствие числа в таблице означает, что прямой дороги между соответствующими пунктами нет.)

	A	B	C	D	E	F
A	-	3	4	-	-	-
B	3	-	-	2	8	-
C	4	-	-	3	6	-
D	-	2	3	-	-	-
E	-	8	6	-	-	5
F	-	-	-	-	5	-

Определите длину кратчайшего пути между пунктами *A* и *F* (при условии, что передвигаться можно только по построенным дорогам).

6. У исполнителя *Вычислитель* имеется три команды:

- 1) **умножь на 3,**
- 2) **умножь на 4,**
- 3) **вычти 5.**

Выполняя первую из них, *Вычислитель* увеличивает число на экране в 3 раза, выполняя вторую, увеличивает число на экране в 4 раза, а выполняя третью, уменьшает число на 5.

Запишите порядок команд в программе преобразования числа 7 в число 44, содержащий не более 4 команд, указывая лишь номера команд.

Например, 2131 — это программа:

умножь на 3, умножь на 4, вычти 5, умножь на 3,
которая преобразует число 1 в число 21.

7. Дан фрагмент электронной таблицы:

	A	B
1	2	= СРЗНАЧ (А3; А4) – 3
2	4	= НОД (А3 : А4)
3		= А4 – 2 * А1 + 2
4	12	= МИН (А1; А4)

Какое число должно быть записано в ячейке А3, чтобы построенная после выполнения вычислений диаграмма по значениям диапазона ячеек В1:В4 соответствовала рисунку 29?

8. Определите значение переменной *c* после выполнения фрагмента программы, записанного ниже на различных языках программирования (переменные *c*, *m*, *d* — целого типа).

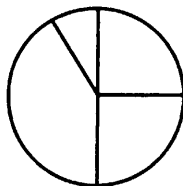


Рис. 29.

Бейсик	Алгоритмический язык
<pre> C = 0 : M = 6312 WHILE M > 10 D=M MOD 10 : C=(C+D)*10 M = M \ 10 WEND C = C + M </pre>	<pre> с := 0; м := 6312 нц пока м > 10 d:=mod(м,10) ; с:=(с+d)*10 м := div(м, 10) кц с := с + м </pre>
Паскаль	Си
<pre> с := 0; м := 6312; while м > 10 do begin d:=м mod 10; с:=(с+d)*10; м := м div 10; end; с := с + м; </pre>	<pre> с = 0; м = 6312; while (м > 10) { d=м%10; с=(с+d)*10; м /= 10; } с = с + м; </pre>

9. Сервер провайдера подключён к Интернету по высокоскоростному каналу 16 Мбит в секунду. Пользователь подключён к Интернету через сервер данного провайдера, причём скорость канала между провайдером и пользователем составляет 512 Кбит в секунду. Пользователю требуется скачать из Интернета файл размером 12 Мб. Сервер провайдера начинает трансляцию данных не раньше, чем им получены первые 1024 Кб этих данных.

Каков минимально возможный промежуток времени (в секундах) с момента начала скачивания провайдером данных до полного их получения пользователем?

10. На одном предприятии работает 236 людей. Для каждого рабочего необходимо завести карточку, номер которой состоял бы из цифр 9, 7, 5, 3. При этом все номера карт должны иметь одну длину и быть различными. Определите минимальную длину номера карты.

11. Ниже на различных языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык
<pre> FUNCTION F(n) IF n = 2 THEN t = -1 ELSE IF n = 1 THEN t = 1 ELSE t = F(n-1) + F(n-2) + 1 END IF END IF PRINT t, F = t END FUNCTION </pre>	<pre> алг цел F(цел n) нач цел t если n=2 то t:=-1 иначе если n=1 то t:=1 иначе t:=F(n-1)+F(n-1)+1 все все вывод t, ' ' знач:=t кон </pre>
Паскаль	Си
<pre> function F(n:integer):integer; var t:integer; begin if n=2 then t:=-1 else if n=1 then t:=1 else t:=F(n-1)+F(n-2)+1 end end write(t, ' '); F:=t end; </pre>	<pre> int F(int n) { int t; if (n==2) t=-1; else if (n==1) t=1; else t=F(n-1)+F(n-2)+1; printf("%d ", t); return t; } </pre>

Чему равно последнее число, напечатанное на экране при выполнении вызова F(6)?

12. Составьте допустимый IP-адрес по следующим фрагментам:

А	Б	В	Г	Д	Е	Ж
43.23	197.2	415	29	6	12	1.58

В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу. (Буквы не должны повторяться.)

13. В некоторой стране автомобильный номер длиной 6 символов составляют из заглавных букв (всего используется 26 различных букв) и десятичных цифр, записанных в любом порядке. Каждый такой номер в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит). Определите объём памяти (в байтах), отводимый этой программой для записи 60 номеров. В ответе запишите только число, слово «байт» писать не нужно.

14. Система команд исполнителя *РОБОТ*, «живущего» в прямоугольном лабиринте на клетчатой плоскости, включает 4 команды-приказа и 4 команды — проверки условия.

Команды-приказы:

вверх, вниз, влево и вправо.

При выполнении любой из этих команд *РОБОТ* перемещается на одну клетку соответственно: вверх, вниз, влево, вправо.

Если *РОБОТ* начнёт движение в сторону находящейся рядом с ним стены, то он разрушится, и программа прервётся.

Другие четыре команды проверяют истинность *условия* отсутствия стены у каждой стороны той клетки, где находится *РОБОТ*:
сверху свободно, снизу свободно, слева свободно и справа свободно.

Цикл

ПОКА <условие>

последовательность команд

конец ПОКА

выполняется, пока условие истинно, иначе происходит переход на выполнение следующей команды.

В конструкции

ЕСЛИ <условие>

ТО команда1

ИНАЧЕ команда2

КОНЕЦ ЕСЛИ

выполняется команда1 (если условие истинно) или команда2 (если условие ложно).

Сколько клеток лабиринта соответствуют требованию, что, начав в ней движение и выполнив предложенную программу, *РОБОТ* уцелеет и остановится в закрашенной клетке (см. рис. 30)?

НАЧАЛО

ПОКА <слева свободно ИЛИ снизу свободно>

ЕСЛИ <снизу свободно>

ТО вниз

ИНАЧЕ влево

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

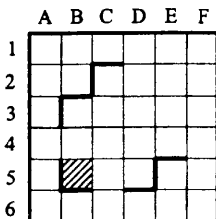


Рис. 30.

1) 1

2) 2

3) 3

4) 10

15. На рисунке 31 изображена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

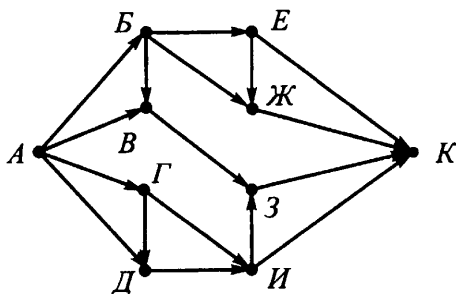


Рис. 31.

16. Сколько цифр в двоичной записи десятичного числа, которое можно представить в виде $4^{360} - 16^{100} - 8^{100}$?

17. В языке запросов поискового сервера для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — символ &.

В таблице приведены запросы и количество страниц, найденных поисковым сервером по этим запросам в некотором сегменте Интернета:

Запрос	Найдено страниц (в тысячах)
цветы & комнатные	672
тюльпан & комнатные	458
(цветы уход) & комнатные	870

Какое количество страниц (в тысячах) будет найдено по запросу «цветы & уход & комнатные»? Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

18. Укажите количество натуральных значений X , при которых истинно логическое выражение: $(X^2 < 25) \wedge ((X < 12) \rightarrow ((X - 3)^2 > 0))$.

19. В программе используется двумерный числовой массив A размером 10×10 , все элементы которого равны 1. Определите значение переменной S после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Алгоритмический язык
<pre> S = 0 FOR I = 0 TO 9 FOR J = 0 TO 9 IF I <= J THEN S = S + A(I, J) ELSE S = S - A(I, J) END IF NEXT J NEXT I </pre>	<pre> S:=0 нц для i от 0 до 9 нц для j от 0 до 9 если i<=j то S:=S+A[i,j] иначе S:=S-A[i,j] все кц кц </pre>
Си	Паскаль
<pre> S = 0 for (i=0; i<=4; i++) for (j=0; j<=4; j++) if (i<=j) S+=A[i][j]; else S-=A[i][j]; </pre>	<pre> S:=0 for i:=0 to 4 do for j:=0 to 4 do if i=j then S:=S+A[i,j] else S:=S-A[i,j]; </pre>

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа A и B . Укажите наименьшее из таких чисел x , при вводе которых алгоритм печатает сначала 3, а потом 5.

Алгоритмический язык	Си
алг нач цел x, A, B ввод x $A := 0; B := 0$ нц пока $x > 0$ $A := A + 1$ если $B < 1 + \text{mod}(x, 10)$ то $B := 1 + \text{mod}(x, 10)$ все $x := \text{div}(x, 10)$ кц вывод A, B кон	<pre>#include <stdio.h> void main() { int x, A, B; scanf ("%d", &x); A = 0; B = 0; while (x > 0) { A = A + 1; if (B < 1 + (x % 10)) B = 1 + (x % 10); x = x / 10; } printf("%d\n%d", A, B); }</pre>

Бейсик	Паскаль
INPUT X $A = 0$ $B = 0$ WHILE $X > 0$ $A = A + 1$ IF $B < 1 + (X \text{ MOD } 10)$ THEN $B = 1 + (X \text{ MOD } 10)$ END IF $X = X \setminus 10$ WEND PRINT A PRINT B	<pre>var x, A, B: integer; begin readln(x); A := 0; B := 0; while x > 0 do begin A := A + 1; if B < 1 + (x mod 10) then B := 1 + (x mod 10); x := x div 10 end; writeln(A); writeln(B) end.</pre>

21. Определите, какое число будет напечатано в результате выполнения следующего алгоритма:

Бейсик	Паскаль
<pre> B = 7: L = 1 FOR S = 1 TO B IF S MOD 2 = 0 THEN L = L * F(S) END IF NEXT S PRINT L FUNCTION F (X) IF X = 0 THEN F = 0 ELSE F = X+F(X-1) END IF END FUNCTION </pre>	<pre> var b,s,L: integer; Function F(x:integer):integer; begin if x = 0 then F:=0 else F:=x+F(x-1); end; BEGIN b:=7; L:=1; for s:=1 to b do if s mod 2 = 0 then L:=L*F(s) writeln(L) END. </pre>
Алгоритмический язык	Си
<pre> алг нач цел b,s,L b:=7; L:=1 нц для s от 1 до b если mod(s,2)=0 то L:=L*F(s) все кц вывод L кон алг цел F(цел x) нач если x=0 то знач:=0 иначе знач := x+F(x-1) все кон </pre>	<pre> #include<stdio.h> int F(int x) { if (x==0) {return 0;} else return (x+F(x-1)); } void main() { int b,s,L; b=7; L=1; for (s=1; s<=b; s++) if (s%2==0) L=L*F(s); printf("%d", L); } </pre>

22. У исполнителя *Уменьшитель* две команды, которым присвоены номера:

1. разделить на 2,
2. вычесть 1.

Первая из них уменьшает число на экране в 2 раза, вторая — уменьшает его на 1. Программа для исполнителя *Уменьшитель* — это последовательность команд.

Определите количество программ, которые число 30 преобразуют в число 7.

23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_{10}$, которые удовлетворяют всем перечисленным ниже условиям?

$$\begin{aligned} ((x_1 \equiv x_2) \wedge (x_3 \equiv x_4)) \vee ((\neg x_1 \equiv x_2) \wedge (\neg x_3 \equiv x_4)) &= 0, \\ ((x_3 \equiv x_4) \wedge (x_5 \equiv x_6)) \vee ((\neg x_3 \equiv x_4) \wedge (\neg x_5 \equiv x_6)) &= 0, \\ ((x_5 \equiv x_6) \wedge (x_7 \equiv x_8)) \vee ((\neg x_5 \equiv x_6) \wedge (\neg x_7 \equiv x_8)) &= 0, \\ ((x_7 \equiv x_8) \wedge (x_9 \equiv x_{10})) \vee ((\neg x_7 \equiv x_8) \wedge (\neg x_9 \equiv x_{10})) &= 0. \end{aligned}$$

Часть 2

24. Требовалось написать программу, при выполнении которой с клавиатуры считывается действительное число x и вычисляется значение функции $y(x)$, заданной условиями:

$$y = \begin{cases} 3x, & \text{если } x \geq 1 \text{ или } x = -2, \\ |x^2 - 5|, & \text{если } x < -2, \\ -8, & \text{в остальных случаях.} \end{cases}$$

Программист торопился и написал программу неправильно.

Паскаль

```
var x, y: real;
begin
  readln(x);
  if (x>=1) or (x=-2) then y := 3*x;
  if x<-2 then y := abs(x*x-5)
  else y := -8;
  write(y)
end.
```

Си
<pre>#include <stdio.h> void main() { float x, y; scanf("%f ", &x); if (x>=1 x=-2) y=3*x; if (x<-2) y=abs(x*x-5); else y=-8; printf("%f", y); }</pre>
Алгоритмический язык
<pre>алг нач вещ x, y ввод x если x>=1 или x=-2 то y:=3*x все если x<-2 то y:=iabs(x*x-5) иначе y:=-8 все вывод y кон</pre>
Бейсик
<pre>INPUT x IF x >= 1 OR x=-2 THEN y = 3*x IF x < -2 THEN y = abs(x*x-5) ELSE y = -8 END IF PRINT y END</pre>

Последовательно выполните следующее:

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при значениях x , принадлежащих различным промежуткам: $(-\infty; -2)$, $(-2; 1)$, $-2 \cup [1; +\infty)$.

В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.», если программа ведёт себя по-разному для разных значений, принадлежащих данному промежутку. В столбце «Выполняемая функция» запишите функцию, для которой вычисляется значение. В последнем столбце укажите «да» или «нет».

2. Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы. (Это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы.)

Промежуток	Усл. 1 $x \geq 1$ или $x = -2$	Усл. 2 $x < -2$	Выполняемая функция	Промежуток обрабатывается верно
$(-\infty; -2)$				
$(-2; 1)$				
$-2 \cup [1; +\infty)$				

25. Дан целочисленный массив из 100 элементов. Элементы массива могут принимать целые значения от 1 до 500. Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести минимальное значение среди элементов массива, которые делят максимальное значение элементов массива без остатка. Гарантируется, что в исходном массиве есть хотя бы один элемент, который делит максимальное значение элементов массива без остатка.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

Бейсик	Си
<pre> CONST N = 100 DIM A(N) AS INTEGER DIM I AS INTEGER DIM MAX AS INTEGER DIM MIN AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END </pre>	<pre> #define N 100 void main(void){ int a[N]; int i, max, min; for (i=0; i<N; i++) scanf("%d", &a[i]); ... } </pre>

Паскаль	Алгоритмический язык
<pre> const N=100 var a: array [1..N] of integer; i, max, min: integer; begin for i := 1 to N do readln(a[i]); ... end.</pre>	<pre> алг нач цел N=100 целтаб a[1:N] цел i, max, min нц для i от 1 до N ввод a[i] кц ... кон</pre>
Естественный язык	
Объявляем массив A из 100 элементов. Объявляем целочисленные переменные I, MAX, MIN. В цикле от 1 до 100 вводим элементы массива A с 1-го по 100-ый. ...	

26. Два участника играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(0, 1)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами (x, y) в одну из трёх точек: или в точку с координатами $(2 * x, y + 4)$, или в точку с координатами $(x + 4, 2 * y)$, или в точку с координатами $(x + 1, y + 1)$. Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами $(0, 0)$ не меньше 15 единиц.

Кто выигрывает при безошибочной игре обоих игроков — игрок, который делает первый ход, или игрок, который делает второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

27. На сортировочной станции данные по каждому товару содержат следующую информацию: код страны отправителя (строка без пробелов, длиной не более 10 символов, начинающаяся с одного или более латинских символов, соответствующих стране), код страны адресата (строка без пробелов, длиной не более 10 символов, начинающаяся с одного или более латинских символов, соответствующих стране), код товара (строка без пробелов, длиной не более 10 символов, начинающаяся с одного или более латинских символов, соответствующих наименованию товара), регистрационный номер (семизначное число). Аббревиатурой товара являет-

ся трёхсимвольный код, состоящий из трёх заглавных латинских букв — первых букв кода страны отправителя, кода страны адресата и кода товара.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка, например Borland Pascal 7.0), которая определит информацию по тем товарам, которые имеют заданную аббревиатуру. Информацию о товаре следует выдать в порядке убывания частоты его встречаемости в списке.

На вход программе в первой строке подаётся аббревиатура товара. Количество товара в списке с такой аббревиатурой не больше 20. Во второй строке находится число N — количество товаров, полученных в результате анализа списка, не все из них подходят под указанную аббревиатуру. Значение N может быть очень велико. В каждой из следующих N строк записано три слова: код страны отправителя, код страны адресата и код соответствующего товара. Слова разделяются одним пробелом. В конце и в начале строки пробелов нет. Все символы в списке записаны заглавными латинскими буквами. Гарантируется, что в списке хотя бы один товар с нужной аббревиатурой есть.

Пример входных данных:

FBT

5

FR742 BEL1254 TR4587 1236547

RU1254 FR4567 GT12454 1236548

FR654 GER4526 LK1245 1236549

FR742 BEL1254 TR4587 1236550

FIN1254 BUL252 TW1247 1236551

Пример выходных данных для приведённого выше примера входных данных:

FR742 BEL1254 TR4587 2

FIN1254 BUL252 TW1247 1

§ 2. Ответы к тестам

Ответы к заданиям части 1 (задания 1–11)

№ вар.	Номер задания										
	1	2	3	4	5	6	7	8	9	10	11
1	3	3	4	4	19	193	1	5	100	229	26
2	2	3	2	5	12	4479	2	–5	290	178	39
3	2	4	3	2	15	21221	15	240	A4	ЛАВА	30
4	3	1	4	1	16	1221	2	120	A45	МАЯК	44
5	2	4	3	1101111	19	24	2	12	90	6	24
6	3	2	3	1100100	14	90	4	12	120	5	22
7	2	3	2	10	28	21121	3	90	A5	415	71
8	2	2	2	9	18	12112	15	18	B17	77	65
9	2	4	3	3	<i>ADFG</i>	12	3	5	384	216	11
10	3	2	4	1	<i>ACDFG</i>	4445	2	16	64	1024	21
11	2	1	4	8	13	1132	12	14	129	5	13
12	3	1	3	2	15	1332	6	2136	192,5	4	5

Отвѣты к заданиям части 1 (задания 12–23)

№ вар.	Номер задания													
	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	AEFG	55	3	12	893	170	36	21	996	80	99	448		
2	EDCA	720	13	11	1196	1280	4	22	159	6	129	7		
3	EGGA	250	7	10	1197	150	17	3	9949	21	15	165		
4	FEBH	300	6	20	1016	160	11	6	1007	5	28	128		
5	DEFG	2048	2	24	322	700	6	47	399	1	110	20		
6	EDCH	2048	7	15	1474	127	7	24	6650	3	84	16		
7	EADG	60	1	24	1702	1330	7	0	24316	2	125	68		
8	BHFE	72	1	18	1112	280	5	15	25329	–5	90	68		
9	DEAA	600	5	13	2	360	100	5	911	6	11	48		
10	FECH	300	7	14	1996	130	101	–17	27913	3	18	5		
11	ADB	180	22	13	530	575	1	85	7777	747	29	192		
12	BAЖ	300	3	11	720	260	3	85	104	630	12	64		

Ответы к заданиям части 2

Вариант 1	
24	1. При вводе последовательности чисел 5 6 9 7 12 программа выведет два числа 27 и 3. 2. Пример последовательности, содержащей числа, кратные трём, для которой программа работает правильно: 7 2 3 5 3. 3. Программа содержит две строки с ошибками (Паскаль): 1) Строка с ошибкой: <code>MinX:=3;</code> Верное исправление: <code>MinX:=1000;</code> 2) Строка с ошибкой: <code>MinX:= i</code> Верное исправление: <code>MinX:= X</code>
25	Пример фрагмента программы на языке Паскаль. <pre> i:=0; k:=0; max:=0; repeat while (a[i]*a[i+1]>0) and (i<=n) do begin i:=i+1; k:=k+1 end while (a[i]*a[i+1]<0) and (i<=n) do i:=i+1; if max<k then max:=k; k:=0; until i=n; writeln(max); </pre>
26	(Более подробное решение подобной задачи с пояснением, каким образом можно найти искомые значения S, представлено в книге «Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий. Решение задач СЗ», с. 5–10.) Пусть S — исходное количество камней в куче.

Вариант 1

1. а) Коля может выиграть первым ходом, если

$S \in \{12, 14, \dots, 22\}$.

Для $S \in \{13, 14, \dots, 20\}$ ему нужно удвоить количество камней в куче. Для $S = 21$ он может либо добавить 2 камня, либо удвоить количество камней. Для $S = 22$ Коля может добавить 1 или 2 камня или удвоить количество камней.

Если количество камней в куче меньше 11, то за один ход нельзя получить кучу, в которой больше 22 камней.

1. б) Саша может выиграть первым ходом (как бы ни играл Коля), если исходно в куче будет $S = 11$ камней. Тогда после первого хода Коли в куче будет или 12 камней, или 13 камней, или 22 камня. В любом случае Саше достаточно удвоить количество камней и выиграть своим первым ходом.

26

2. Возможные значения S : 10 и 9. В этих случаях Коля, очевидно, не может выиграть первым ходом. Однако, добавив в кучу 1 или 2 камня, он может получить кучу из 11 камней. Эта позиция разобрана в п. 1 б). Если в куче 11 камней, то игрок, который будет ходить (теперь это Саша), выиграть не может, а его противник (то есть Коля) следующим ходом выиграет.

3. Возможное значение S : 8. После первого хода Коли в куче будет или 9, или 10, или 16 камней. Если в куче станет 16 камней, Саша удвоит количество камней и выиграет первым ходом. Ситуация, когда в куче 10 или 9 камней, разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Саша), выигрывает своим вторым ходом. В таблице изображено дерево возможных партий при описанной стратегии Саши.

Вариант 1

Заключительные позиции (в них выигрывает Саша) подчеркнуты.

26

И.п.	Положение после очередных ходов			
	1-й ход Коли (разобраны все ходы)	1-й ход Саши (только ход по стратегии)	2-й ход Коли (разобраны все ходы)	2-й ход Саши (только ход по стратегии)
8	$8 + 1 = 9$	$9 + 2 = 11$	$11 + 1 = 12$	$12 \cdot 2 = 24$
			$11 + 2 = 13$	$13 \cdot 2 = 26$
			$11 \cdot 2 = 22$	$22 \cdot 2 = 44$
	$8 + 2 = 10$	$10 + 1 = 11$	$11 + 1 = 12$	$12 \cdot 2 = 24$
			$11 + 2 = 13$	$13 \cdot 2 = 26$
			$11 \cdot 2 = 22$	$22 \cdot 2 = 44$
	$8 \cdot 2 = 16$	$16 \cdot 2 = 32$		

27

Программа читает все входные данные один раз. При прочтении данных очередной книги сразу проверяются условия: издана ли эта книга ранее 1980 года и содержит ли она не менее 300 страниц. Если оба условия одновременно выполняются, то значение счётчика увеличивается на единицу и проверяется условие: является ли название такой книги меньше уже просмотренных. В результате прочтения данных обо всех книгах становятся известны количество искомых книг и самое короткое название.

Пример программы на языке Паскаль.

```

var c: Char;
    s, ms: string;
    i, y, p, N, k, ml: Integer;
begin
    Readln(N);
    ml := 41; k := 0;
    for i:=1 to N do
        begin
            repeat
                Read(c);
            until c=' '; {считана фамилия}
        end
    end

```

Вариант 1	
27	<pre> Read(y); Read(p); Readln(s); if (y < 1980) and (p >= 300) then begin Inc(k); if Length(s) < ml then begin ml := Length(s); ms := s; end; end; end; Writeln(k); Writeln(ms) end.</pre>
Вариант 2	
24	1. При вводе последовательности чисел –6 5 –8 4 2 программа выведет два числа –1.6 и 4. 2. Пример последовательности, содержащей чётные числа, для которой программа работает правильно: 4 –8 2 –6 10. 3. Программа содержит три строки с ошибками (Паскаль): 1) Строка с ошибкой: <code>MaxX:=0;</code> Верное исправление: <code>MaxX:=-1000;</code> 2) Строка с ошибкой: <code>if Sum <> 0 then begin</code> Верное исправление: <code>if K > 0 then begin</code> 3) Строка с ошибкой: <code>writeln(Sum/N:2:1);</code> Верное исправление: <code>writeln(Sum/K:2:1);</code>

Вариант 2

Пример фрагмента программы на языке Паскаль.

```
i:=0; k:=0; max:=0;
repeat
  l:=0;
  while (a[i] mod 2 = 0) and (i<=n) do begin
    k:=k+a[i];
    i:=i+1;
    l:=l+1
  end;
  if l=1 then k:=0;
  while (a[i] mod 2 <> 0) and (i<=n) do
    i:=i+1;
  if max<k then
    max:=k;
  k:=0;
until i>n;
writeln(max);
```

(Более подробное решение подобной задачи с пояснением, каким образом можно найти искомые значения S , представлено в книге **«Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий. Решение задач С3»**, с. 5–10.)

Пусть S — исходное количество камней в куче.

1. а) Коля может выиграть первым ходом, если

$S \in \{21, 22, \dots, 40\}$. Для $S \in \{21, 22, \dots, 38\}$ ему нужно удвоить количество камней в куче. Для $S = 39$ он может либо добавить 2 камня, либо удвоить количество камней. Для $S = 40$ Коля может добавить 1 или 2 камня, или удвоить количество камней. Если количество камней в куче меньше 21, то за один ход нельзя получить кучу, в которой больше 40 камней.

1. б) Саша может выиграть первым ходом (как бы ни играл Коля), если исходно в куче будет $S = 20$ камней. Тогда после первого хода Коли в куче будет или 21 камень, или 22 камня, или 40 камней. В любом случае Саше достаточно удвоить количество камней и выиграть своим первым ходом.

Вариант 2

26

2. Возможные значения S : 19 и 18. В этих случаях Коля, очевидно, не может выиграть первым ходом. Однако, добавив в кучу 1 или 2 камня, он может получить кучу из 20 камней. Эта позиция разобрана в п. 1 б). Если в куче 20 камней, то игрок, который будет ходить (теперь это Саша), выиграть не может, а его противник (то есть Коля) следующим ходом выигрывает.

3. Возможное значение S : 17. После первого хода Коли камней в куче будет или 18, или 19, или 34. Если в куче станет 34 камня, Саша удвоит количество камней и выиграет первым ходом. Ситуация, когда в куче 18 или 19 камней, разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Саша), выигрывает своим вторым ходом. В таблице изображено дерево возможных партий при описанной стратегии Саши. Заключительные позиции (в них выигрывает Саша) подчеркнуты.

И.п.	Положение после очередных ходов			
	1-й ход Коли (разобраны все ходы)	1-й ход Саши (только ход по стратегии)	2-й ход Коли (разобраны все ходы)	2-й ход Саши (только ход по стратегии)
17	$17 + 1 = 18$	$18 + 2 = 20$	$20 + 1 = 21$	<u>$21 \cdot 2 = 42$</u>
			$20 + 2 = 22$	<u>$22 \cdot 2 = 44$</u>
			$20 \cdot 2 = 40$	<u>$40 \cdot 2 = 80$</u>
	$17 + 2 = 19$	$19 + 1 = 20$	$20 + 1 = 21$	<u>$21 \cdot 2 = 42$</u>
			$20 + 2 = 22$	<u>$22 \cdot 2 = 44$</u>
			$20 \cdot 2 = 40$	<u>$40 \cdot 2 = 80$</u>
	$17 \cdot 2 = 34$	<u>$34 \cdot 2 = 68$</u>		

27

Пример программы на языке Turbo Pascal 7.0

```
var Nmar, Id, tpm, Strf: array[1..100] of integer;
    Predpr: array[1..100] of string;
    M, N, i, j, t, Interv, Nm: integer;
begin
    write(' M = ');
    readln(M);
    writeln(' Nmar Id Predpr');
```

Вариант 2

27

```
for i:=1 to M do
    readln(Nmar[i],Id[i],Predpr[i]);
for i:=1 to M do
begin
    tpm[i]:=0; Strf[i]:=0;
end;
write(' N = ');
readln(N);
writeln(' t Nmar');
for j:=1 to N do begin
    readln(t,Nm);
    for i:=1 to M do
        if Nm = Nmar[i] then break;
    Interv:=t-tpm[i]-Id[i];
    if Interv > 2 then
        Strf[i]:= Strf[i]+Interv;
    tpm[i]:=t;
end;
for i:=1 to M do
if tpm[i]<60 then
begin
    Interv:=60-tpm[i]-Id[i];
    if Interv > 2 then
        Strf[i]:= Strf[i]+Interv;
end;
for i:=1 to M do
    write(Nmar[i], ' ',Id[i], ' ',
    writeln(Strf[i], ' ',Predpr[i]);
readln;
end.
```

Вариант 3	
24	1. При вводе чисел 5426 и 7318 программа выведет число 10. 2. Пример входных чисел, для которых программа работает правильно: 1154 и 2531. 3. Программа содержит две строки с ошибками (Паскаль): 1) Строку $\text{MaxN}:=0; \text{MaxM}:=0;$ следует заменить на $\text{MaxN}:=N \bmod 10; \text{MaxM}:=M \bmod 10;$ 2) Строку $\text{MaxM}:=N \bmod 10;$ следует заменить на $\text{MaxM}:=M \bmod 10;$ 4. Для улучшения эффективности работы программы строку $\text{while } (N > 10) \text{ or } (M > 10) \text{ do begin}$ следует заменить на $\text{while } N > 10 \text{ do begin}$
25	Пример фрагмента программы на языке Паскаль. <pre> p := 1; for i := 0 to N-1 do if (i mod 2 = 0) and (a[i] mod 7=0) then p:=p*a[i]; writeln(p); </pre>
26	(Более подробное решение подобной задачи с пояснением, каким образом можно найти искомые значения S, представлено в книге «Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий. Решение задач С3», с. 5–10.) Пусть S — исходное количество камней в куче. 1. а) Коля может выиграть первым ходом, если $S \in \{37, 38, \dots, 108\}$.

Вариант 3

Для $S \in \{37, 38, \dots, 107\}$ ему нужно утроить количество камней в куче. Для $S = 108$ он может либо добавить 1 камень, либо утроить количество камней.

Если количество камней в куче меньше 37, то за один ход нельзя получить кучу, в которой больше 108 камней.

1. 6) Саша может выиграть первым ходом (как бы ни играл Коля), если исходно в куче будет $S = 36$ камней. Тогда после первого хода Коли в куче будет или 37 камней, или 108 камней. В любом случае Саше достаточно утроить количество камней и выиграть своим первым ходом.

2. Возможные значения S : 12 и 35. В этих случаях Коля, очевидно, не может выиграть первым ходом. Однако, добавив в кучу с 35 камнями 1 камень или утроив кучу из 12 камней, он может получить кучу из 36 камней. Эта позиция разобрана в п. 1 б). Если в куче 36 камней, то игрок, который будет ходить (теперь это Саша), выиграть не может, а его противник (то есть Коля) следующим ходом выигрывает.

3. Возможное значение S : 34. После первого хода Коли в куче будет или 35 камней, или 102 камня. Если в куче станет 102 камня, Саша утроит количество камней и выиграет первым ходом. Ситуация, когда в куче 35 камней, разобрана в п. 2.

В этой ситуации игрок, который будет ходить (теперь это Саша), выигрывает своим вторым ходом.

В таблице изображено дерево возможных партий при описанной стратегии Саши. Заключительные позиции (в них выигрывает Саша) подчеркнуты.

И.п.	Положение после очередных ходов			
	1-й ход Коли (разобраны все ходы)	1-й ход Саши (только ход по стратегии)	2-й ход Коли (разобраны все ходы)	2-й ход Саши (только ход по стратегии)
34	$34 + 1 = 35$	$35 + 1 = 36$	$36 + 1 = 37$	<u>$37 \cdot 3 = 111$</u>
			$36 \cdot 3 = 108$	<u>$108 + 1 = 109$</u>
	$34 \cdot 3 = 102$	<u>$102 \cdot 3 = 306$</u>		

Вариант 3

Программа читает все входные данные один раз, сразу подсчитывая в массиве с индексами от 1 до 20 количество камней, принадлежащих определённой ценовой категории. Путём просмотра этого массива с конца (от 20-й ценовой категории) определяется число камней, заведомо попадающих в число 5% самых дорогих (добавление всех камней следующей ценовой категории приводит к выходу за 5%). Последняя ценовая категория, в которую попало не менее одного драгоценного камня, запоминается. Если хотя бы один из камней следующей ценовой категории также попадает в 5%, то проверяется, имеют ли он и другие камни, набравшие столько же баллов, ценовую категорию не менее 15. В этом случае они все добавляются к числу драгоценных камней, которые необходимо поместить в сейф и ценовая категория которых является искомой.

27 Пример программы на языке Turbo Pascal 7.0.:

```
var kcenkat : array[1..20] of integer;
c : char; i,N,b,S,minckat : integer;
begin
  for i:=1 to 20 do kcenkat[i]:=0;
  readln(N);
  for i:=1 to N do
  begin
    repeat
      read(c);
    until c=' '; {считано название драгоценного камня}
    repeat
      read(c);
    until c=' '; {считан код}
    readln(b);
    kcenkat[b]:=kcenkat[b]+1;
  end;
```

Вариант 3

27 S:=0; b:=20;
 while S+kcenkat[b]<=N*0.05 do
 begin
 if kcenkat[b]>0 then
 begin
 S:=S+kcenkat[b]; minckat:=b;
 end;
 b:=b-1;
 end; {определены те камни, которые наверняка попада-
 ют в сейф, и пропущены ценовые категории, в которые
 не попал ни один камень}
 if S+1<=N*0.05 then
 {если ещё хотя бы один драгоценный камень попадёт
 в 5%, то проверяется: ценовая категория этого и
 таких же камней — не менее 15}
 if b>=15 then minckat:=b;
 writeln(minckat);
 end.

Вариант 4

24 1. При вводе чисел 517 и 834 программа выведет число 8.
 2. Пример входных чисел, для которых программа работает
 правильно: 135 и 423.
 3. Программа содержит две строки с ошибками (Паскаль):
 1) Строку
 S:= (N + M) mod 10;
 следует заменить на
 S:=N mod 10 + M mod 10;
 2) Строку
 while N > 0 do begin
 следует заменить на
 while N > 10 do begin
 4. Для улучшения эффективности работы программы нуж-
 но убрать строку
 while M > 10 do

Вариант 4

25

Пример фрагмента программы на языке Паскаль.

```
sum := 0; k:=0;
for i := 0 to N-1 do
begin
  if (a[i] > 0) and (a[i] mod a[0] = 0) then
  begin
    sum := sum + a[i];
    k := k + 1
  end
end;
writeln(sum/k);
```

26

(Более подробное решение подобной задачи с пояснением, каким образом можно найти искомые значения S , представлено в книге «Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий. Решение задач СЗ», с. 5–10.)

Пусть S — исходное количество камней в куче.

1. а) Коля может выиграть первым ходом, если $S \in \{16, 17, \dots, 28\}$.

Во всех случаях ему нужно удвоить количество камней в куче.

Если количество камней в куче меньше 16, то за один ход нельзя получить кучу, в которой больше 30 камней.

1. б) Саша может выиграть первым ходом (как бы ни играл Коля), если исходно в куче будет $S = 14$ камней. Тогда после первого хода Коли в куче будет или 16 камней, или 28 камней. В любом случае Саше достаточно удвоить количество камней и выиграть своим первым ходом.

2. Возможные значения S : 12 и 7. В этих случаях Коля, очевидно, не может выиграть первым ходом. Однако, добавив в кучу с 12 камнями 2 камня или удвоив кучу из 7 камней, он может получить кучу из 14 камней. Эта позиция разобрана в п. 1 б). Если в куче 14 камней, то игрок, который будет ходить (теперь это Саша), выиграть не может, а его противник (то есть Коля) следующим ходом выигрывает.

Вариант 4

26

3. Возможное значение S : 10. После первого хода Коли в куче будет или 12 камней, или 20 камней. Если в куче станет 20 камней, Саша удвоит количество камней и выиграет первым ходом. Ситуация, когда в куче 12 камней, разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Саша), выигрывает своим вторым ходом. В таблице изображено дерево возможных партий при описанной стратегии Саши. Заключительные позиции (в них выигрывает Саша) подчеркнуты.

И.п.	Положение после очередных ходов			
	1-й ход Коли (разобраны все ходы)	1-й ход Саши (только ход по стратегии)	2-й ход Коли (разобраны все ходы)	2-й ход Саши (только ход по стратегии)
10	$10 + 2 = 12$	$12 + 2 = 14$	$14 + 2 = 16$	<u>$16 \cdot 2 = 32$</u>
			$14 \cdot 2 = 28$	<u>$28 \cdot 2 = 56$</u>
	$10 \cdot 2 = 20$	<u>$20 \cdot 2 = 40$</u>		

27

Заметим, что искомое контрольное число может быть получено в результате произведения двух чисел, обладающих свойствами:

1) оба числа кратны 14; первое из чисел наибольшее среди всех элементов, второе наибольшее из всех, кроме первого (согласно условию, числа могут быть равными);

2) одно число — наибольшее из чисел, кратных 14, другое — наибольшее из чисел, не кратных 14 (но, возможно, кратных 2 или 7);

3) одно число — наибольшее из чисел, кратных 2, другое — наибольшее из чисел, кратных 7, при этом каждое из чисел не кратно 14.

Таким образом, реализация алгоритма состоит из трёх этапов.

1 этап. В процессе ввода элементов определяем значения четырёх переменных Max_{14} , Max , Max_2 и Max_7 :

Max_{14} — наибольшее из чисел, кратных 14;

Max — наибольшее из чисел, за исключением Max_{14} ;

Max_2 — наибольшее из чисел, не кратных 14, но кратных 2;

Max_7 — наибольшее из чисел, не кратных 14, но кратных 7.

Вариант 4

2 этап. Поиск наибольшего произведения среди $\text{Max14} * \text{Max}$ и $\text{Max2} * \text{Max7}$.

3 этап. Определяем совпадение найденного произведения с контрольным значением.

Пример программы на языке Turbo Pascal 7.0

```
var x, Max14, Max, Max2, Max7, N, i: longint;
```

```
    R, R_max : longint;
```

```
begin
```

```
{1 этап}
```

```
    Max14:=0; Max:=0; Max2:=0; Max7:=0;    readln(N);
```

```
    for i:=1 to N do begin
```

```
        readln(x);
```

```
        if x mod 14 <> 0 then begin
```

```
            if x>Max then Max:=x;
```

```
            if (x mod 2 = 0) and (x>Max2) then Max2:=x
```

```
            else
```

```
                if (x mod 7 = 0) and (x>Max7) then Max7:=x
```

```
        end
```

```
    else
```

```
        if x>Max14 then Max14:=x
```

```
        else if x>Max then Max:=x;
```

```
    end;
```

```
    readln(R);
```

```
{2 этап}
```

```
    if (Max14*Max > Max2*Max7) then
```

```
        R_Max:=Max14*Max
```

```
    else R_Max:=Max2*Max7;
```

```
{3 этап}
```

```
    if R=R_Max then
```

```
        writeln('Контроль пройден')
```

```
    else
```

```
        writeln('Контроль не пройден')
```

```
end.
```

27

Вариант 5	
24	1. При вводе чисел 615 и 214 программа выведет 615 и 4 соответственно. 2. Пример входного числа, для которого программа работает правильно: 5. 3. Программа содержит две строки с ошибками (Паскаль): 1) Операторные скобки поставлены ошибочно. Строку <pre>if N mod 2 = 0 then begin</pre> следует заменить на <pre>if N mod 2 = 0 then</pre> Строку <pre>end;</pre> следует убрать. 2) Строку <pre>while N > 0 do</pre> следует заменить на <pre>while N > 10 do</pre>
25	Пример фрагмента программы на языке Паскаль. <pre>sum := 0; for i := 0 to N-1 do if (a[i] >= 10) and (a[i] <= 99) and (a[i] mod 10 + a[i] div 10) < 10) then sum := sum + a[i]; writeln(sum);</pre>
26	Выигрывает Миша. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке набором из двух цифр записано количество камней в кучках на каждом этапе игры. В таблице записаны все возможные первые ходы обоих игроков и выигрышные ходы (один из возможных в каждой позиции) первого игрока после первого хода. Обратим также внимание на то, что последовательность кучек не важна, так что их можно переставлять, и это не повлияет на доказательство.

Вариант 5

26

Начальная ситуация	Ход Миши	Ход Вовы	Ход Миши
7 7	8 8	9 9	
		10 10	
		10 8	
		8 10	
	9 9	10 10	12 10
		11 11	12 12
		9 11	10 12
		11 9	12 10
	9 7	10 8	
		11 9	
		9 9	
		11 7	
	7 9	9 9	
		7 11	
		8 10	
		9 11	

Из таблицы видно, что последовательность безошибочных ходов (с учётом минимального их количества) выглядит следующим образом: 7 7 → 9 9 → 11 11 → 12 12. При этом Вова не выбывает из соревнований.

27

Пример программы на языке Turbo Pascal 7.0.

```

var fio : array[1..100] of string;
    bs : array[1..100] of integer;
    max,N,i,k,b : integer; c : char;
begin
    readln(N);
    max:=0;
    for i:=1 to N do
        begin
            fio[i]:='';

```

Вариант 5

27

```

for k:=1 to 2 do
  repeat
    read(c);
    fio[i]:=fio[i]+c;
  until c=' ';
read(bs[i]);
for k:=1 to 2 do
begin
  read(b);
  bs[i]:=bs[i]+b;
end;
if bs[i]>max then
  max:=bs[i];
end;
for i:=1 to N do
  if bs[i]=max then
    writeln(fio[i],max);
end.

```

Вариант 6

24

1. При вводе числа 4715 программа выведет 4.
2. Пример входного числа, для которого программа работает правильно: 7468
3. Программа содержит две строки с ошибками (Паскаль):
 - 1) Строку


```
if VAL < 5 then
```

следует заменить на

```
if VAL <= 5 then
```
 - 2) Строку


```
S:=VAL
```

следует заменить на

```
S:=S+VAL
```

Вариант 6

Пример фрагмента программы на языке Паскаль.

25

```
kol_p := 0; kol_m := 0;
for i := 1 to N do begin
  if (a[i] > 0) then kol_p := kol_p + 1;
  if (a[i] < 0) then kol_m := kol_m + 1 end;
writeln(kol_p - kol_m);
```

Выигрывает первый игрок, Георгий. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке набором из двух цифр записано количество камней в кучках на каждом этапе игры.

В таблице записаны все возможные первые ходы обоих игроков и выигрышные ходы (один из возможных в каждой позиции) первого игрока после первого хода. Обратим также внимание на то, что последовательность кучек не важна, так как их можно переставлять, и это не повлияет на доказательство.

26

Начальная ситуация	Ход Георгия	Ход Валерия	Ход Георгия
5 7	7 7	9 7	
		7 9	
		11 7	
		7 11	
	5 9	9 9	13 9
		7 9	7 11
		5 11	7 11
		5 13	9 13
	5 11	7 11	
		9 11	
		5 13	
		5 15	
	9 7	11 7	
		13 7	
		9 9	
		9 11	

Из таблицы видно, что последовательность безошибочных ходов (с учётом минимального их количества) может выглядеть следующим образом: 5 7 → 5 9 → 5 11 → 7 11. При этом победитель получит 18 пирожков.

Вариант 6

Пример программы на языке Turbo Pascal 7.0.

```
type userData = record
    login, ip_add : string;
end;
var N, i, j, k : integer;
    dt : array [1..100] of userData;
    t : userData; c : char;
    subnetwork, log, ip : string; flag : boolean;
begin
    writeln('Введите адрес сети (подсети)');
    readln(subnetwork);
    writeln('Введите N'); readln(N);
    writeln('Введите ', N, ' строк данных');
    j := 0;
    for i := 1 to N do
        begin
            log := '';
            repeat
                read(c);
            until c = ' ';
            repeat
                read(c); log := log + c;
            until c = ' ';
            readln(ip);
            if Pos(subnetwork,ip) = 1 and
                (ip[length(subnetwork)+1]='.') then begin
                j := j + 1;
                dt[j].login := log; dt[j].ip_add := ip
            end;
        end;
    k := j;
```

Вариант 6

27

```
repeat
  flag := false;
  for i := 1 to k - 1 do begin
    if dt[i].login > dt[i+1].login then begin
      t := dt[i]; dt[i] := dt[i+1];
      dt[i+1] := t; flag := true;
    end;
  end;
  k := k - 1;
until not flag;
for i := 1 to j do
  writeln(dt[i].login + ' ' + dt[i].ip_add);
end.
```

Вариант 7

24

1. При вводе числа 4357 программа выведет два числа 12 и 3.
 1. Пример входного числа, для которого программа работает правильно: 718.
 2. Программа содержит две строки с ошибками (Паскаль):
 1) Строку
 K:= K+VAL;
 следует заменить на
 K:= K+1;
 2) Строку
 if MIN > 0 then begin
 следует заменить на
 if K > 0 then begin

25

Пример фрагмента программы на языке Паскаль.
 max:=0; k:=0;
 for i := 1 to N do
 if (1000>a[i]) and (99<a[i]) and (max<a[i])
 and (a[i] mod 5 = 0) then

Вариант 7

25

```

begin
    max:= a[i];  k:=k+1
end;
if k>0 then
    then writeln(max)
else
    writeln(' Не найдено ');

```

26

Выигрывает второй игрок, своим первым ходом он должен поставить фишку в точке с координатами $(-1; 4)$. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке приведены координаты фишки на каждом этапе игры.

1 ход	2 ход	3 ход	4 ход
I-й игрок (все варианты хода)	II-й игрок (выигрыш- ный ход)	I-й игрок (все варианты хода)	I-й игрок (выигрыш- ный ход)
-1; 3	-1; 4	-1; 6	-1; 12
		-1; 5	-1; 10
		-1; 8	-1; 10
-1; 2	-1; 4	Те же варианты 3-го — 4-го ходов.	
-1; 2	Те же варианты 2-го — 4-го ходов		

Таблица содержит все возможные варианты ходов первого игрока. Из неё видно, что при любом ответе первого игрока у второго имеется ход, приводящий к победе.

27

Пример программы на языке Turbo Pascal 7.0

```

function get_data(s, data : string) : string;
var  p, p1 : integer;
    s1 : string;
begin
    p:=Pos(data,s);
    if p<>0 then

```

Вариант 7

```

begin
    s1:=Copy(s,p+length(data),length(s)-5);
    p1:=Pos(' ', s1);
    if p1=0 then get_data:=s1
    else get_data:=Copy(s,p+length(data),p1-1);
end
else get_data:='';
end;
var N, i, p : integer;
    ms : array [0..999] of string;
    s, time, src, dst : string;
    flag : boolean;
begin
    readln(N);
    for i := 0 to N - 1 do readln(ms[i]);
    readln(s);
    time:=get_data(s,'time=');
    src:=get_data(s,'src=');
    dst:=get_data(s,'dst=');
    for i := 0 to N - 1 do begin
        flag:=true;
        if time<>'' then begin
            p:=Pos(time,ms[i]);
            if p<>1 then flag:=false;
        end;
        p:=Pos(' ',ms[i]);
        s:=Copy(ms[i],p+1,length(ms[i])-p);
        if src<>'' then
            begin
                p:=Pos(src,s);
                if p<>1 then flag:=false;
            end;
    end;

```

27

Вариант 7	
27	<pre> p:=Pos(' ',s); s:=Copy(s,p+1,length(s)-p); if dst<>' ' then begin p:=Pos(dst,s); if p<>1 then flag:=false; end; if flag then writeln(ms[i]); end; end.</pre>
Вариант 8	
24	1. При вводе числа 713 программа выведет 9. 2. Пример входного числа, для которых программа работает правильно: 925. 3. Программа содержит две строки с ошибками (Паскаль): 1) Строку MAX:=9; следует заменить на MAX:=0; 2) Строку while N > 10 do следует заменить на while N > 0 do
25	Пример фрагмента программы на языке Паскаль. <pre> min:=999; k:=0; for i := 1 to N do if (1000>a[i]) and (99<a[i]) and (min>a[i]) and (a[i] mod 3 = 0) then begin min:= a[i]; k:=i end; if k>0 then writeln(k) else writeln(' Не найдено ');</pre>

Вариант 8

26

Выигрывает первый игрок. Его первый ход — в точку с координатами (5; 2). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

Для первого игрока указаны выигрышные ходы, для второго — все варианты ходов, возможные после очередного хода первого игрока. Жирным шрифтом выделены последние ходы победителя.

Старт	1-й ход	2-й ход	3-й ход
	1 игрок	2 игрок	1 игрок
(3; 2)	(5; 2)	(6; 2)	(8; 2)
		(7; 2)	(8; 2)
		(5; 4)	(7; 4)

27

Программа читает все входные данные один раз, не запоминая их в массиве, а составляя только список встретившихся регионов и количество заявок по каждому из этих регионов, которое хранится в массиве *cnt*. Если при обработке данных находится регион, в котором несколько заводов подают заявку, то программа каждый раз автоматически увеличивает количество заявок от региона на 1. После окончания обработки данных программа производит поиск самого низкого числа заявок.

После нахождения самого низкого числа заявок программа проверяет, сколько всего имеется регионов с таким же количеством заявок. В конце выводятся коды регионов с самым низким числом заявок.

Пример программы на языке Turbo Pascal 7.0.

```
var  cnt: array[1..99] of integer;
     c: char; i, r, N, min: integer;
     f: boolean;
begin
  for i := 1 to 99 do cnt[i] := 0;
  readln(N); {считано количество заявок}
```

Вариант 8

27

```
for i := 1 to N do
begin
  repeat
    read(c);
  until c = ' '; {считано название завода}
  repeat
    read(c);
  until c = ' '; {считан номер лицензии завода}
  readln(r);
  cnt[r] := cnt[r] + 1
end;
min := 0;
f := true;
{производим поиск минимального элемента в массиве cnt}
for i := 1 to 99 do begin
  if (cnt[i] < min) and (cnt[i] <> 0) then
    min := cnt[i];
  if f and (cnt[i] <> 0) then begin
    min := cnt[i];
    f := false
  end
end;
{выводим коды регионов с наименьшим количеством заявок}
writeln('регион(ы) с наименьшим числом заявок:')
for i := 1 to 99 do
  if cnt[i] = min then
    writeln(i)
end.
```

Вариант 9

24	1. При вводе последовательности чисел 3 2 8 7 4 программа выведет два числа 2 и 4. 2. Пример последовательности, содержащей чётные числа, для которой программа работает правильно: 1 2 4 5 6. 3. Программа содержит две строки с ошибками (Паскаль): 1) Строки: <pre> if x < MinX then MinX:=X else MaxX:= X </pre> Следует заменить на: <pre> if X < MinX then MinX:=X; if X < MaxX then MaxX:=X; </pre> 2) Строку: <pre> if MinX <> 0 then begin </pre> следует заменить на : <pre> if MinX <> 1000 then begin </pre>
25	Пример фрагмента программы на языке Паскаль. <pre> prod := 1; for i := 0 to N-1 do if (a[i] >= 50) and (a[i] <= 79) and (a[i] mod 2 = 0) then prod := prod * a[i]; writeln(prod); </pre>
26	Будем описывать количество камней в кучках набором из трёх цифр. Например, начальное положение описывается набором 3 3 2. Обратим также внимание, что последовательность кучек не важна, так как их можно переставлять, что не влияет на доказательство. У первого игрока есть две выигрышные стратегии, выбор зависит от первого хода первого игрока.

Вариант 9

Первым ходом первый игрок должен взять из первой (или второй) кучки 2 камня (получится 1 3 2) или из третьей 2 (получится 3 3 0). Дальнейшее поведение первого игрока в зависимости от действий второго опишем следующими двумя таблицами (для первого игрока указан один из возможных ходов в каждой позиции, для второго — все возможные в данной позиции ходы):

26

Ситуация после первого хода 1-го игрока	2-й игрок	1-й игрок	2-й игрок	1-й игрок
1 3 2	0 3 2	0 2 2	0 1 2	1 0 0
	1 2 2		0 0 2	
	1 3 1	1 1 1	0 1 1	
	1 1 2			
	1 3 0	→		
	1 0 2			
Ситуация после первого хода 1-го игрока	2-й игрок	1-й игрок	2-й игрок	1-й игрок
3 3 0	2 3 0	2 2 0	2 1 0	1 0 0
			2 0 0	
	1 3 0	→		
	0 3 0			

27

Программа читает все входные данные один раз, сразу подсчитывая в массиве с индексами от 0 до 50 количество участников, набравших тот или иной балл. Путём просмотра этого массива с конца (от 50 баллов) определяется число участников, заведомо попадающих в число 45% лучших (добавление всех участников, набравших следующий балл, приводит к выходу за 45%).

Последний балл, который набрали не менее одного участника, запоминается. Если хотя бы один из следующих участников также попадает в 45%, то проверяется, набрал ли он и следующие, набравшие столько же баллов, более половины баллов. В этом случае они все добавляются к числу победителей и призёров, и их балл является искомым.

Вариант 9

27

```

Пример программы на языке Turbo Pascal 7.0.
var  cnt: array[0..50] of integer;
c: char; i, k, N, b, S, minb: integer;
begin
  for i:=0 to 50 do cnt[i]:=0;
  readln(N);
  for i:=1 to N do begin
    repeat
      read(c);
    until c=' '; {считана фамилия}
    repeat
      read(c);
    until c=' '; {считано имя}
    readln(k,b); cnt[b]:=cnt[b]+1;
  end;
  S:=0; b:=50;
  while (S + cnt[b])*100<=N*45 do begin
    S:=S+cnt[b];
    if cnt[b]>0 then minb:=b;
    b:=b-1
  end; {определены те, кто наверняка стал призѐром, и про-
пушены баллы, которые никто не набрал}
  if (S+1)*100<=N*45 then
    {если ещё хотя бы один участник попадает в 45%, то прове-
ряется, какой балл набрал он и следующие участники}
    if b>25 then minb:=b
    writeln(minb)
end.

```

Вариант 10

24

1) Пример входных данных, для которых программа рабо-
тает неправильно: $x = 1$, $y = 1$. Для этих координат про-
грамма выдаёт сообщение **НЕ ПРИНАДЛЕЖИТ**, хотя точка при-
надлежит заданной области, поскольку находится на её гра-
нице.

Вариант 10

24) Пример правильной программы на языке Паскаль:

```

var x, y: real;
begin
  readln(x,y);
  if ((y<=1) and (y>=0) and (x>=0) and (x<=1)) or
    ((x>=-1) and (x<=0) and (y>=0) and
    (x*x+y*y<=1)) then
    writeln('ПРИНАДЛЕЖИТ')
  else
    writeln('НЕ ПРИНАДЛЕЖИТ')
end.
```

25) Пример фрагмента программы на языке Паскаль.

```

sum := 0;
for i := 0 to N-1 do
  if (a[i] >= 10) and (a[i] <= 99) and
    (a[i] mod 10 <> 3) then
    sum := sum + a[i];
writeln(sum);
```

26) Выигрывает второй игрок. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записана длина куска верёвки, оставшегося после хода игрока. Для первого игрока указаны все варианты ходов, для второго — выигрышные ходы.

1-й ход	2-й ход	3-й ход	4-й ход
I-й игрок	II-й игрок	I-й игрок	II-й игрок
18 – 4 = 14	14 – 5 = 9	9 – 4 = 5	5 – 5 = 0
		9 – 5 = 4	4 – 4 = 0
18 – 5 = 13	13 – 4 = 9	9 – 4 = 5	5 – 5 = 0
		9 – 5 = 4	4 – 4 = 0
	13 – 5 = 8	8 – 4 = 4	4 – 4 = 0
		8 – 5 = 3	3 – 3 = 0

Таблица содержит все возможные варианты ходов первого игрока. Из неё видно, что при любом ходе первого игрока у второго имеется ход, приводящий к победе.

Вариант 10

Пример программы на языке Turbo Pascal 7.0.

```
const F = 64;
var cnt: array[1..F] of real; c : char;
    z, max, sum: real; i, flt, N: integer;
begin
    for i := 1 to F do cnt[i] := 0;
    readln(N);
    max:=0; sum:=0;
    for i := 1 to N do begin
        repeat
            read(c);
        until c = ' '; {конец фамилии}
        repeat
            read(c);
        until c = ' '; {конец имени}
        readln(flt, z); {номер кв-ры, задолженность}
        if z > max then max := z;
        sum := sum+z; cnt[flt] := z
    end;
    flt := 0; i := 1;
    if max = sum/N then
        while flt <= (N*0.6) do begin
            if cnt[i] > 0 then begin
                write (i, ' '); flt := flt + 1
            end;
            i := i + 1
        end
    else
        for i := 1 to F do
            if cnt[i] > max*0.8 then write (i, ' ');
    end.
```

Вариант 11

1.

Проме- жуток	Усл. 1 $x > 2$	Усл. 2 $-1 \leq x \leq 2$	Выполняемая функция	Промежуток обрабаты- вается верно
$(-\infty; -1)$	нет	нет	$y = x $	да
$[-1; 2]$	нет	да	$y = 2x^2 - 3$	да
$(2; +\infty)$	да	нет	$y = x $	нет

2. Возможная доработка программы на языке Паскаль:

24

```

var x, y: real;
begin
  readln(x);
  if x>2 then y:=1/x
  else
    if (x>=-1) and (x<=2) then y:=2*x*x-3
    else y:=abs(x);
  write(y)
end.

```

Возможны и другие способы доработки.

Пример фрагмента программы на языке Паскаль.

25

```

max := 55;
for i:=1 to N do
  if (a[i] mod 2=1) and (a[i] mod 9<>0) and (a[i]>max)
  then max := a[i];
writeln(max);

```

26

Выигрывает второй игрок. Его первый ход — в точку с координатами (2; 4) или (4; 3). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры. Для первого игрока указаны все возможные в данной позиции ходы, для второго — выигрышные ходы (один из возможных в каждой позиции).

Вариант 11

26

Старт	1 ход	2 ход	3 ход	4 ход
	1 игрок	2 игрок	1 игрок	2 игрок
(-2; 1)	(2; 1)	(2; 4)	(6; 4)	(10; 4)
			(2; 7)	(6; 7)
			(4; 6)	(8; 6)
	(-2; 4)	(2; 4)	(6; 4)	(10; 4)
			(2; 7)	(6; 7)
			(4; 6)	(8; 6)
	(0; 3)	(4; 3)	(8; 3)	(12; 3)
			(4; 6)	(8; 6)
			(6; 5)	(10; 5)

После четвёртого хода игра останавливается, так как координаты фишки $(x; y)$ удовлетворяют условию $x^2 + y^2 > 9$.

27

Программа читает все входные данные один раз, не запоминая их в массиве, а запоминая только тип операции над акцией и цену, соответствующую данной акции. Программа производит покупку акции, если тип операции с ней соответствует покупке и если цена покупки не превышает приемлемой цены 240. При покупке акции увеличивается количество и общая стоимость всех приобретённых акций. В конце выводится число всех приобретённых акций, а также средняя стоимость одной приобретённой акции.

Пример программы на языке Turbo Pascal 7.0.

```
var i, k, N, p: integer; s: real; c: char;
begin
  readln(N); {считано количество акций}
  s := 0; {общая стоимость приобретённых акций}
  k := 0; {количество приобретённых акций}
  for i := 1 to N do
    begin
      repeat
        read(c);
      until c = ' '; {считано название акции}
```

Вариант 1

27

```

read(c); {считан тип операции}
if c = 'b' then begin
  read(c); readln(p);
  {сравниваем текущую цену с приемлемой ценой 240}
  if p <= 240 then begin
    s := s + p; {покупаем акцию}
    k := k + 1; {увеличиваем число имеющихся акций}
  end;
end;
end;
writeln('совершено ',k,'операций по покупке акций');
writeln('средняя стоимость 1 приобр. акции - ',s/k);
end.

```

Вариант 12

1.

Проме- жуток	Усл. 1 $x \geq 1$ или $x = -2$	Усл. 2 $x < -2$	Выпол- няемая функция	Промежуток обрабаты- вается верно
$(-\infty; -2)$	нет	да	$y = x^2 - 5 $	да
$(-2; 1)$	нет	нет	$y = -8$	да
$-2 \cup [1; +\infty)$	да	нет	$y = -8$	нет

24

2. Возможная доработка программы на языке Паскаль:

```

var x, y: real;
begin
  readln(x);
  if (x >= 1) or (x = -2) then
    y := 3*x
  else
    if x < -2 then y := abs(x*x-5)
    else y := -8;
  write(y)
end.

```

Возможны и другие способы доработки.

Вариант 12

25

Пример фрагмента программы на языке Паскаль.

```

max := 1;
for i := 1 to N do
  if (a[i] > max)
  then
    max := a[i];
min := 500;
for i := 1 to N do
  if (a[i] < min) and (max mod a[i] = 0)
  then min := a[i];
writeln(min);
    
```

26

Выигрывает первый игрок. Его первый ход — в точку с координатами (4; 2). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры. Для первого игрока указаны выигрышные ходы (один из возможных в каждой позиции), для второго — все возможные в данной позиции ходы.

Старт	1 ход	2 ход	3 ход	4 ход	5 ход
(0; 1)	(4; 2)	(8; 6)	(16; 10)		
		(8; 4)	(16; 8)		
		(5; 3)	(6; 4)	(12; 8)	(24; 12)
				(10; 8)	(20; 12)
				(7; 5)	(14; 9)

В зависимости от второго хода второго игрока игра останавливается либо после третьего хода, либо после пятого хода, поскольку координаты фишки $(x; y)$ при правильной игре первого игрока удовлетворяют условию $x^2 + y^2 > 15$.

Вариант 12

Пример программы на языке Turbo Pascal 7.0.

```

var N, k, k1, i, j, L : integer;
{Массив для хранения информации о товаре }
    Tov: array[1..20] of string;
{Массив для хранения информации о количестве одинакового товара }
    Kol : array [1..20] of integer;
    abr:string[3];
    s1,s:string[40];
begin
    writeln('Введите аббревиатуру');
    readln(abr);
    writeln('Введите количество элементов списка');
    readln(N);
    L := 0; i:=1;
27 {Считываем очередную информацию о товаре и проверяем совпаде-
    ние первой буквы с первой буквой аббревиатуры. Если нет совпаде-
    ния, считываем следующие данные списка}
    while i<=N do
    begin
        readln(s);
        if s[1]<>abr[1] then i:=i+1
        else
        begin
            s1:=s; k:=pos(' ',S);
            delete(s1,1,k); k1:=pos(' ',S1);
            {Проверяем на совпадение с заданной аббревиатурой.}
            if (s[k+1]=abr[2]) and (s[k+k1+1]=abr[3]) then
            begin
                S:=copy(s,1,length(s)-8);
            {Осуществляем поиск данных о товаре среди введённых ранее и под-
            считываем количество совпадающей информации}

```

Вариант 12

```

j := 1;
while (j<=L) and (s<>Tov[j])
    do j:=j+1;
if j<=L then  Kol[j]:=Kol[j]+1
else begin
    Tov[j]:=s; Kol[j]:=1; L:=L+1
end
end;
i:=i+1
end
end;

```

{Выводим на экран значения массивов Tov и Kol в порядке убывания значений массива Kol. Для этого, начиная с первого элемента, находим наибольший элемент массива Kol и выводим на экран этот элемент и соответствующий ему элемент массива Tov. На место найденного элемента ставим первый. Далее находим наибольший, начиная со второго элемента массива Kol, и т.д.}

27

```

for i:=1 to L-1 do
begin
    n:=kol[i]; k:=i;
    for j:=i+1 to L do
begin
    if n<kol[j] then begin
        n:=kol[j]; k:=j;
    end;
end;
writeln('Tov=',Tov[k], ' k=',Kol[k]);
Kol[k]:=Kol[i];
Tov[k]:=Tov[i];
end;
writeln('Tov=',Tov[L], ' k=',Kol[L]);
end.

```

Литература

1. *Андреева Е.В., Босова Л.Л., Фалина И.Н.* Математические основы информатики. Элективный курс: учебное пособие — 2-е изд., испр. и доп. — М.: БИНОМ, Лаборатория знаний, 2007.
2. *Евич Л.Н.* Информатика и ИКТ. Экспресс-курс. Подготовка к ЕГЭ. Пособие с электронным приложением (CD-диск). / Под редакцией С.Ю. Кулабухова. — Ростов-на-Дону: Легион, 2014.
3. *Евич Л.Н.* Информатика и ИКТ. Подготовка к ЕГЭ. Системы счисления. / Под редакцией С.Ю. Кулабухова. — Ростов-на-Дону: Легион, 2014.
4. *Евич Л.Н.* Информатика и ИКТ. Сборник задач по программированию. Подготовка к ЕГЭ. / Под редакцией С.Ю. Кулабухова. — Ростов-на-Дону: Легион, 2014.
5. *Евич Л.Н.* Информатика и ИКТ. Подготовка к ЕГЭ. Элементы алгебры логики. / Под редакцией С.Ю. Кулабухова. — Ростов-на-Дону: Легион, 2014.
6. *Евич Л.Н.* Информатика и ИКТ. Подготовка к ЕГЭ. Поиск выигрышных стратегий. / Под редакцией С.Ю. Кулабухова. — Ростов-на-Дону: Легион, 2014.
7. *Костюк Ю.Л., Фукс И.Л.* Основы разработки алгоритмов: учебное пособие. — М.: БИНОМ, Лаборатория знаний, 2010.
8. *Кочетов Ю.А.* Дискретная математика.
<http://www.math.nsc.ru/LBRT/k5/dm.html> [Электронный ресурс], 2009.

9. Олимпиады по базовому курсу информатики : методическое пособие / под ред. С.В. Русакова. — 2-е изд. — М.: БИНОМ, Лаборатория знаний, 2009.
10. Поляков К.Ю. <http://kpolyakov.narod.ru/> [Электронный ресурс]
11. Угринович Н.Д. Информатика и ИКТ. Профильный уровень: учебник для 10 класса / 9-е изд., испр. и доп. — М.: БИНОМ, Лаборатория знаний, 2013.
12. Угринович Н.Д. Информатика и ИКТ. Профильный уровень: учебник для 11 класса / 4-е изд., испр. и доп. — М.: БИНОМ, Лаборатория знаний, 2012.

Готовимся к ЕГЭ

Учебное издание

Евич Людмила Николаевна

**ИНФОРМАТИКА И ИКТ
ПОДГОТОВКА К ЕГЭ-2015**

Пособие с электронным приложением (CD-диск)

Под редакцией *Л. Н. Евич, С. Ю. Кулабухова*

Налоговая льгота: издание соответствует коду 95 3000 ОК 005-93 (ОКП)

Обложка *А. Вартанов*

Компьютерная верстка *Л. Евич*

Корректоры *Л. Андреева*

Подписано в печать с оригинал-макета 09.10.2014.

Формат 60х84¹/₁₆. Бумага типографская.

Гарнитура Таймс. Печать офсетная. Усл. печ. л. 15,81.

Тираж 5000 экз. Заказ № **32**.

Издательство ООО «Легион» включено в перечень организаций, осуществляющих издание учебных пособий, которые допускаются к использованию в образовательном процессе в имеющих государственную аккредитацию и реализующих образовательные программы общего образования образовательных учреждениях. Приказ Минобрнауки России № 729 от 14.12.2009, зарегистрирован в Минюст России 15.01.2010 № 15987.

ООО «ЛЕГИОН»

Для писем: 344000, г. Ростов-на-Дону, а/я 550.

Адрес редакции: 344011, г. Ростов-на-Дону, пер. Долгомановский, 55.

www.legionr.ru e-mail: legionrus@legionrus.com

Отпечатано в соответствии с качеством предоставленных
диапозитивов в ООО «Полиграфобъединение»

347900, г. Таганрог, ул. Лесная биржа, 6в.