

**Тренировочная работа № 4**

**по ИНФОРМАТИКЕ**

**17 мая 2013 года**

**11 класс**

**Вариант ИНФ1601**

**Район**

---

**Город (населённый пункт)**

---

**Школа**

---

**Класс**

---

**Фамилия**

---

**Имя**

---

**Отчество**

---

**Инструкция по выполнению работы**

На выполнение экзаменационной работы по информатике и ИКТ отводится 235 минут. Экзаменационная работа состоит из 3 частей, содержащих 32 задания. Рекомендуем не более 1,5 часов (90 минут) отвести на выполнение заданий частей 1 и 2, а остальное время – на часть 3

Часть 1 содержит 13 заданий (A1–A13). К каждому заданию даётся четыре варианта ответа, из которых только один правильный

Часть 2 состоит из 15 заданий с кратким ответом (B1–B15). К этим заданиям Вы должны самостоятельно сформулировать и записать ответ

Часть 3 состоит из 4 заданий (C1–C4). Для выполнения заданий этой части Вам необходимо написать развёрнутый ответ в произвольной форме

Все бланки ЕГЭ заполняются яркими чёрными чернилами. Допускается использование гелевой, капиллярной или перьевой ручек

При выполнении заданий Вы можете пользоваться черновиком. Обращаем Ваше внимание на то, что записи в черновике не будут учитываться при оценивании работы

Советуем выполнять задания в том порядке, в котором они даны. Для экономии времени пропускайте задание, которое не удаётся выполнить сразу, и переходите к следующему. Если после выполнения всей работы у Вас останется время, Вы сможете вернуться к пропущенным заданиям

Баллы, полученные Вами за выполненные задания, суммируются. Постарайтесь выполнить как можно больше заданий и набрать наибольшее количество баллов

*Желаем успеха!*

В экзаменационных заданиях используются следующие соглашения.

1. Обозначения для логических связок (операций):

а) *отрицание* (инверсия, логическое НЕ) обозначается  $\neg$  (например,  $\neg A$ );

б) *конъюнкция* (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );

в) *дизъюнкция* (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );

г) *импликация* (следование) обозначается  $\rightarrow$  (например,  $A \rightarrow B$ );

д) *тождество* обозначается  $\equiv$  (например,  $A \equiv B$ ); выражение  $A \equiv B$  истинно тогда и только тогда, когда значения  $A$  и  $B$  совпадают (либо они оба истинны, либо они оба ложны);

е) символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).

2. Два логических выражения, содержащие переменные, называются *равносильными* (эквивалентными), если значения этих выражений совпадают при любых значениях переменных. Так, выражения  $A \rightarrow B$  и  $(\neg A) \vee B$  равносильны, а  $A \vee B$  и  $A \wedge B$  – нет (значения выражений разные, например, при  $A = 1, B = 0$ ).

3. Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование). Таким образом,  $\neg A \wedge B \vee C \wedge D$  совпадает с  $((\neg A) \wedge B) \vee (C \wedge D)$ . Возможна запись  $A \wedge B \wedge C$  вместо  $(A \wedge B) \wedge C$ . То же относится и к дизъюнкции: возможна запись  $A \vee B \vee C$  вместо  $(A \vee B) \vee C$ .

4. Везде в задачах 1 Кбайт = 1024 байт, 1 Мбайт = 1024 Кбайт.

### Часть 1

При выполнении заданий этой части в бланке ответов № 1 под номером выполняемого Вами задания (A1–A13) поставьте знак «X» в клеточке, номер которой соответствует номеру выбранного Вами ответа.

**A1** Какое из приведённых выражений имеет наибольшее значение?

1)  $128_{10} + 64_{10} + 8_{10} + 4_{10}$

2)  $313_8$

3)  $E5_{16}$

4)  $11100111_2$

**A2** Между населёнными пунктами A, B, C, D, E, F, Z построены дороги, протяжённость которых приведена в таблице. (Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.)

	A	B	C	D	E	F	Z
A		4	10	15			34
B	4		4	6			
C	10	4		2			
D	15	6	2		3	11	15
E				3		8	9
F				11	8		4
Z	34			15	9	4	

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

1) 34

2) 30

3) 25

4) 22

**A3** Дан фрагмент таблицы истинности выражения F:

x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	F
0	1	0	1	1	1	0	1	1	1	1
1	0	1	1	0	0	1	1	1	0	1
0	1	0	1	0	1	0	0	1	0	0

Каким выражением может быть F?

1)  $(x1 \vee \neg x2) \wedge (x3 \vee \neg x4) \wedge x5 \wedge \neg x6 \wedge x7 \wedge x8 \wedge \neg x9 \wedge x10$

2)  $(x1 \wedge \neg x2) \vee (x3 \wedge \neg x4) \vee x5 \vee \neg x6 \vee x7 \vee x8 \vee \neg x9 \vee x10$

3)  $(\neg x1 \wedge x2) \vee (\neg x3 \wedge x4) \vee \neg x5 \vee x6 \vee \neg x7 \vee \neg x8 \vee x9 \vee \neg x10$

4)  $(\neg x1 \vee x2) \wedge (\neg x3 \vee x4 \wedge \neg x5 \wedge x6 \wedge \neg x7 \wedge \neg x8 \wedge x9 \wedge \neg x10)$

**A4** 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 находится 6 файлов:

inmar.xls  
komarik.xlsx  
komarik.xxx  
marka.xlsx  
romario.xls  
smart.xlsx

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

komarik.xlsx  
marka.xlsx  
romario.xls  
smart.xlsx

1) \*?mar\*.x\*

2) ?mar\*.xls\*

3) \*?mar\*.xls\*

4) \*mar?\*.xls\*

**A5** Автомат получает на вход трёхзначное десятичное число, в котором все цифры нечётные. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также вторая и третья цифры.
2. Полученные два числа записываются друг за другом в порядке возрастания (без разделителей).

**Пример.** Исходное число: 571. Суммы:  $5+7 = 12$ ;  $7+1 = 8$ . Результат: 812.

Определите, какое из перечисленных ниже чисел может быть результатом работы автомата.

1) 148

2) 417

3) 816

4) 914

**A6** Во фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите, сколько всего родных братьев и сестёр есть у Штольц Т.И.

ID	Фамилия_И.О.	Пол
1465	Дядюн М.Б.	Ж
1493	Баль А.П.	М
1560	Штольц И.Б.	М
1625	Рерих А.И.	Ж
1837	Штольц П.И.	М
1851	Радек П.А.	Ж
1885	Штольц Б.Ф.	М
1983	Чиж Д.К.	Ж
2216	Рерих Л.А.	Ж
2226	Штольц А.Б.	Ж
2398	Малеев К.Г.	М
2470	Баль П.А.	М
2607	Штольц Т.И.	Ж
2737	Панина Р.Г.	Ж
2759	Тесленко Г.Р.	Ж
2788	Рерих В.Б.	Ж
...	...	...

ID_Родителя	ID_Ребёнка
1493	2470
1560	1837
1560	2607
1885	1465
1885	1560
1885	2226
1885	2788
1983	1465
1983	1560
1983	2226
1983	2788
2226	2470
2759	1837
2759	2607
2788	1851
2788	2216
...	...

1) 1

2) 2

3) 3

4) 0

**A7** В ячейке E16 электронной таблицы записана формула. Эту формулу скопировали в ячейки D17 и C18. В соответствии с формулой, полученной в ячейке D17, значение в этой ячейке равно сумме значений в ячейках D31 и C32; в соответствии с формулой, полученной в ячейке C18, значение в этой ячейке равно сумме значений в ячейках D32 и B32.

Укажите, какая формула могла быть написана в ячейке E16.

*Примечание: знак \$ используется для обозначения абсолютной адресации.*

1) =\$D30 + D\$32

2) =\$D\$32 + \$B\$32

3) =\$D\$31 + \$C\$32

4) =E\$32 + \$D30

**A8** Производилась четырёхканальная (квадро) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. В результате был получен файл размером 60 Мбайт, сжатие данных не производилось. Какая из приведённых ниже величин наиболее близка ко времени, в течение которого проводилась запись?

- 1) 1 мин.      2) 2 мин.      3) 3 мин.      4) 4 мин.

**A9** По каналу связи передаются сообщения, содержащие только 4 буквы: Е, Н, О, Т.

В любом сообщении больше всего букв О, следующая по частоте буква – Е, затем – Н. Буква Т встречается реже, чем любая другая.

Для передачи сообщений нужно использовать неравномерный двоичный код, допускающий однозначное декодирование; при этом сообщения должны быть как можно короче. Шифровальщик может использовать один из перечисленных ниже кодов. Какой код ему следует выбрать?

- 1) Е – 0, Н – 1, О – 00, Т – 11      2) О – 1, Н – 0, Е – 01, Т – 10  
3) Е – 1, Н – 01, О – 001, Т – 000      4) О – 0, Н – 11, Е – 101, Т – 100

**A10** На числовой прямой даны два отрезка:  $P = [30, 50]$  и  $Q = [10, 70]$ . Выберите такой отрезок А, чтобы формула

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

была тождественно истинна, то есть принимала значение 1 при любом значении переменной  $x$ . Если таких отрезков несколько, укажите тот, который имеет меньшую длину.

- 1) [27, 33]      2) [27, 53]      3) [7, 33]      4) [7, 53]

**A11** При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 6 символов и содержащий только символы из 7-буквенного набора Н, О, Р, С, Т, У, Х. В базе данных для хранения сведений о каждом пользователе отведено одинаковое целое число байт, при этом для хранения сведений о 100 пользователях используется 1400 байт. Для каждого пользователя хранятся пароль и дополнительные сведения. Для хранения паролей используют посимвольное кодирование, все символы кодируются одинаковым и минимально возможным количеством бит. Сколько бит отведено для хранения дополнительных сведений о каждом пользователе?

- 1) 88      2) 90      3) 94      4) 98

**A12** Ниже представлен записанный на разных языках программирования фрагмент одной и той же программы. В программе описан одномерный целочисленный массив А, в представленном фрагменте программы обрабатываются элементы массива с индексами от 1 до 10.

<b>Бейсик</b>	n = 10 FOR i = 1 TO n A(n+1-i) = 2*A(i) NEXT i
<b>Паскаль</b>	n := 10; for i := 1 to n do begin A[n+1-i] := 2*A[i]; end;
<b>Си</b>	n = 10; for (i = 1; i <= n; i++) A[n+1-i] = 2*A[i];
<b>Алгоритмический язык</b>	n := 10 <u>нц</u> <u>для</u> i <u>от</u> 1 <u>до</u> n A[n+1-i] := 2*A[i] <u>кц</u>

Перед началом выполнения фрагмента элементы массива имеют значения соответственно

$$1, 2, 4, 8, 16, 32, 64, 128, 256, 512, \text{ т.е. } A[k] = 2^{k-1}, k = 1, \dots, 10.$$

Укажите значение, которое после выполнения указанного фрагмента программы имеют два или более рассмотренных в этом фрагменте элемента массива. Если таких чисел несколько, укажите наименьшее из них.

- 1) такого значения нет      2) 2  
3) 8      4) 4

**A13** Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости, состоит из 8 команд. Четыре команды – это команды-приказы:

<b>вверх</b>	<b>вниз</b>	<b>влево</b>	<b>вправо</b>
--------------	-------------	--------------	---------------

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:

<b>сверху свободно</b>	<b>снизу свободно</b>	<b>слева свободно</b>	<b>справа свободно</b>
------------------------	-----------------------	-----------------------	------------------------

Цикл

ПОКА *условие*

*последовательность команд*

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ *условие*

ТО *команда1*

ИНАЧЕ *команда2*

КОНЕЦ ЕСЛИ

выполняется *команда1* (если условие истинно) или *команда2* (если условие ложно).

В конструкциях ПОКА и ЕСЛИ *условие* может содержать команды проверки, а также слова И, ИЛИ, НЕ, обозначающие логические операции.

Если РОБОТ начнёт движение в сторону находящейся рядом с ним стены, то он разрушится, и программа прервётся.

Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка F6)?

НАЧАЛО

ПОКА **снизу свободно ИЛИ справа свободно**

ЕСЛИ **снизу свободно**

ТО

**вниз**

КОНЕЦ ЕСЛИ

ЕСЛИ **справа свободно**

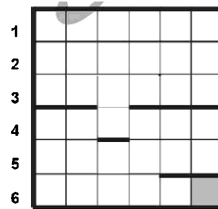
ТО

**вправо**

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ



A B C D E F

1) 12

2) 16

3) 20

4) 24

## Часть 2

*Ответом к заданиям этой части (B1–B15) является число, последовательность букв или цифр. Впишите ответы сначала в текст работы, а затем перенесите их в бланк ответов № 1 справа от номера соответствующего задания, начиная с первой клеточки, без пробелов, запятых и других дополнительных символов. Каждую букву или цифру пишите в отдельной клеточке в соответствии с приведёнными в бланке образцами.*

**B1** У исполнителя Аккорд две команды, которым присвоены номера:

1. отними 1

2. умножь на 5

Выполняя первую из них, Аккорд отнимает от числа на экране 1, а выполняя вторую, умножает это число на 5.

Запишите порядок команд в программе, которая содержит не более 5 команд и переводит число 5 в число 98.

В ответе указывайте лишь номера команд, пробелы между цифрами не ставьте. Так, для программы

умножь на 5

отними 1

отними 1

нужно написать: 211. Эта программа преобразует, например, число 4 в число 18.

Ответ:

- В2** Определите значение переменной **c** после выполнения следующего фрагмента программы (записанного ниже на разных языках программирования). Ответ запишите в виде целого числа.

<b>Бейсик</b>	<pre>a = 40 b = 12 a = a/5 - b/3 IF 4*a &gt; b THEN   c = a - 3*b + 37 ELSE   c = a + 3*b + 37 END IF</pre>
<b>Паскаль</b>	<pre>a := 40; b := 12; a := a/5 - b/3; if 4*a &gt; b then   c := a - 3*b + 37 else   c := a + 3*b + 37;</pre>
<b>Си</b>	<pre>a = 40; b = 12; a = a/5 - b/3; if (4*a &gt; b)   c = a - 3*b + 37; else   c = a + 3*b + 37;</pre>
<b>Алгоритмический</b>	<pre>a := 40 b := 12 a := a/5 - b/3 если 4*a &gt; b   то c := a - 3*b + 37   иначе c := a + 3*b + 37 все</pre>

Ответ:

- В3** Дан фрагмент электронной таблицы:

	A	B	C
1	20		48
2	=C1-B1*B1*5	=2*(B1*B1*B1+3)/A1	=C1-15*B1

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



Известно, что все значения диапазона A1:C2 имеют один и тот же знак.

Ответ:

- В4** На световой панели в ряд расположены 7 лампочек. Каждая из первых двух лампочек может гореть красным, жёлтым или зелёным цветом. Каждая из остальных пяти лампочек может гореть одним из двух цветов – красным или белым. Сколько различных сигналов можно передать с помощью панели (все лампочки должны гореть, порядок цветов имеет значение)?

Ответ:

- В5** Определите, что будет напечатано в результате выполнения программы (записанной ниже на разных языках программирования):

<b>Бейсик</b>	<pre>DIM N, S AS INTEGER N = 30 S = 1 WHILE S &lt; 500   S = S * 2   N = N + 10 WEND PRINT N</pre>
---------------	--

<b>Паскаль</b>	<pre>var n, s: integer; begin   n := 30;   s := 1;   while s &lt; 500 do   begin     s := s * 2;     n := n + 10;   end;   write(n); end.</pre>
<b>Си</b>	<pre>#include&lt;stdio.h&gt; void main() {   int n, s;   n = 30;   s = 1;   while (s &lt; 500)   {     s = s * 2;     n = n + 10;   }   printf("%d", n); }</pre>
<b>Алгоритмический</b>	<pre><u>алг</u> <u>нач</u>   <u>цел</u> n, s   n := 30   s := 1   <u>нц пока</u> s &lt; 500     s := s * 2     n := n + 10   <u>кц</u>   <u>вывод</u> n <u>кон</u></pre>

Ответ:

**В6** Алгоритм вычисления значения функции  $F(n)$ , где  $n$  – натуральное число, задан следующими соотношениями:

$F(1) = 1;$

$F(n) = 2 * F(n-1) + 1$  при  $n > 1$ .

Чему равно значение функции  $F(5)$ ?

*В ответе запишите только натуральное число.*

Ответ:

**В7** Запись числа  $N$  в системе счисления с основанием 6 содержит две цифры, запись этого числа в системе счисления с основанием 5 содержит три цифры, а запись в системе счисления с основанием 11 заканчивается на 1. Чему равно  $N$ ?

Ответ:

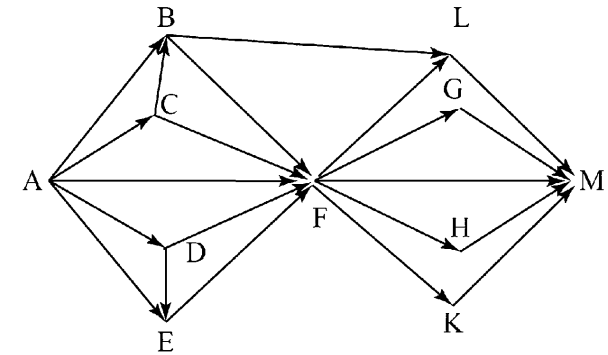
**В8** Ниже на четырёх языках записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа  $a$  и  $b$ . Укажите наибольшее из таких чисел  $x$ , при вводе которых алгоритм печатает сначала 2, а потом 13.

<b>Бейсик</b>	<pre>DIM X, A, B AS INTEGER INPUT X A = 0: B = 0 WHILE X &gt; 0   A = A+1   B = B + (X MOD 100)   X = X\100 WEND PRINT A PRINT B</pre>
<b>Паскаль</b>	<pre>var x, a, b: integer; begin   readln(x);   a := 0; b := 0;   while x &gt; 0 do   begin     a := a+1;     b := b + (x mod 100);     x := x div 100;   end;   writeln(a); write(b); end.</pre>

<b>Си</b>	<pre>#include &lt;stdio.h&gt; void main() {     int x, a, b;     scanf("%d", &amp;x);     a = 0; b = 0;     while (x &gt; 0) {         a = a+1;         b = b + (x%100);         x = x/100;     }     printf("%d\n%d", a, b); }</pre>
<b>Алгоритмический</b>	<pre><u>алг</u> <u>нач</u>     <u>цел</u> x, a, b     <u>ввод</u> x     a:=0; b:=0     <u>нц пока</u> x &gt; 0         a := a+1         b := b+mod(x,100)         x := div(x,100)     <u>кц</u>     <u>вывод</u> a, <u>нс</u>, b <u>кон</u></pre>

Ответ:

- В9** На рисунке – схема дорог, связывающих города A, B, C, D, E, F, G, H, K, L, M. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города A в город M?



Ответ:

- В10** Документ (без упаковки) можно передать по каналу связи с одного компьютера на другой за 75 секунд. Если предварительно упаковать документ архиватором, передать упакованный документ, а потом распаковать на компьютере получателя, то общее время передачи (включая упаковку и распаковку) составит 30 секунд. При этом на упаковку и распаковку данных всего ушло 15 секунд. Размер исходного документа 20 Мбайт. Чему равен размер упакованного документа (в Мбайт)? В ответе запишите только число.

Ответ:



**В11** 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 заданным IP-адресу узла и маске.

По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 224.34.225.134

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 без использования точек.

A	B	C	D	E	F	G	H
255	254	244	224	134	34	8	0

Пример.

Пусть искомым 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.

Ответ:

**В12** В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» – символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тыс.)
Леннон & Маккартни & Старт	1100
Леннон & Маккартни & Харрисон	1300
Леннон & Маккартни & Харрисон & Старт	1000

Какое количество страниц (в тыс.) будет найдено по запросу

(Леннон & Маккартни & Старт) | (Леннон & Маккартни & Харрисон)?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ:

**В13** У исполнителя Полтора две команды, которым присвоены номера:

1. прибавь один,

2. умножь на полтора.

Первая из них увеличивает на 1 число на экране, вторая увеличивает это число в 1,5 раза, если число чётное. К нечётным числам вторая команда неприменима.

Программа для Полтора – это последовательность команд.

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

Ответ:

**В14** Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

<b>Бейсик</b>	<pre> DIM A, B, T, M, R AS INTEGER A = -7: B = 25 M = A: R = F(A) FOR T = A TO B+3     IF F(T) &lt; R THEN         M = T         R = F(T)     END IF NEXT T PRINT M FUNCTION F(x)     F = 182 - 2*(19+x)*(19+x) END FUNCTION </pre>
---------------	---

<b>Паскаль</b>	<pre> var a,b,t,M,R :integer; Function F(x: integer):integer; begin     F := 182 - 2*(19+x)*(19+x); end; BEGIN a := -7; b := 25; M := a; R := F(a); for t := a to b+3 do begin     if (F(t) &lt; R) then begin         M := t;         R := F(t);     end; end; write(M); END. </pre>
----------------	---

<b>Си</b>	<pre>#include&lt;stdio.h&gt; int F(int x) {     return 182 - 2*(19+x)*(19+x); } void main() {     int a, b, t, M, R;     a = -7; b = 25;     M = a; R = F(a);     for (t=a; t&lt;=b+3; t++){         if (F(t) &lt; R) {             M = t; R = F(t);         }     }     printf("%d", M); }</pre>
-----------	---

<b>Алгоритмический</b>	<pre>алг нач     цел a, b, t, R, M     a := -7; b := 25     M := a; R := F(a)     нц для t от a до b+3         если F(t) &lt; R             то                 M := t; R := F(t)         все     кц     вывод M кон алг цел F(цел x) нач     знач := 182 - 2*(19+x)*(19+x) кон</pre>
------------------------	--

Ответ:

**B15** 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?

$$(x_1 \rightarrow x_2) \wedge (x_2 \rightarrow x_3) \wedge (x_3 \rightarrow x_4) \wedge (x_4 \rightarrow x_5) = 1$$

$$(y_2 \rightarrow y_1) \wedge (y_3 \rightarrow y_2) \wedge (y_4 \rightarrow y_3) \wedge (y_5 \rightarrow y_4) = 1$$

$$x_1 \vee y_1 = 1$$

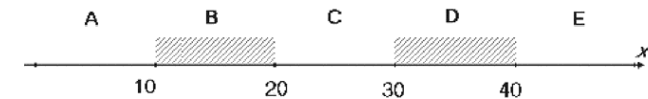
В ответе **не нужно** перечислять все различные наборы значений переменных  $x_1, x_2, x_3, x_4, x_5, y_1, y_2, y_3, y_4, y_5$ , при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

Ответ:

### Часть 3

Для записи ответов к заданиям этой части (C1–C4) используйте бланк ответов № 2. Запишите сначала номер задания (C1 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

**C1** Требовалось написать программу, при выполнении которой с клавиатуры считывается координата точки на прямой ( $x$  – действительное число) и определяется принадлежность этой точки заданной области (включая границы). Программист торопился и написал программу неправильно.



#### Бейсик

```
INPUT x
IF x<=30 THEN
    IF x<=20 THEN
        IF x<=10 THEN
            PRINT "не принадлежит"
        ELSE
            PRINT "принадлежит"
        END IF
    END IF
END IF
END IF
END
```

**Паскаль**

```

var x: real;
begin
  readln(x);
  if x<=30 then
    if x<=20 then
      if x<=10 then
        write('не принадлежит')
      else
        write('принадлежит')
    end.
end.

```

**Си**

```

#include <stdio.h>
void main()
{
  float x;
  scanf("%f", &x);
  if (x<=30)
    if (x<=20)
      if (x<=10)
        printf("не принадлежит");
      else
        printf("принадлежит");
}

```

**Алгоритмический язык**

```

алг
нач
  вещ x
  ввод x
  если x<=30 то
    если x<=20 то
      если x<=10 то
        вывод 'не принадлежит'
      иначе
        вывод 'принадлежит'
    все
  все
все
кон

```

Последовательно выполните следующее.

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (А, В, С, D и Е). Точки, лежащие на границах областей (то есть, точки 10, 20, 30 и 40) при заполнении таблицы не рассматриваются.

Область	Условие 1 ( $x \leq 30$ )	Условие 2 ( $x \leq 20$ )	Условие 3 ( $x \leq 10$ )	Программа выведет	Область обрабатывается верно
А					
В					
С					
D					
Е					

В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.», если программа ведет себя по-разному для разных значений, принадлежащих данной области. В столбце «Программа выведет» укажите, что программа выведет на экран. Если программа ничего не выводит, поставьте «—» (прочерк). Если для разных значений, принадлежащих области, будут выведены разные тексты, напишите «не изв.». 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.)

**C2** Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Элемент массива называется хорошим, если это двузначное число, причём цифра в разряде десятков больше, чем цифра в разряде единиц. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести сумму всех хороших элементов массива.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

#### Паскаль

```
const
    N=30;
var
    a: array [1..N] of integer;
    i, j, s: integer;
begin
    for i:=1 to N do
        readln(a[i]);
    ...
end.
```

#### Бейсик

```
N=30
DIM A(N) AS INTEGER
DIM I, J, S AS INTEGER
FOR I = 1 TO N
    INPUT A(I)
NEXT I
...
END
```

#### Си

```
#include <stdio.h>
#define N 30
void main(){
    int a[N];
    int i, j, s;
    for (i=0; i<N; i++)
        scanf("%d", &a[i]);
    ...
```

#### Алгоритмический язык

```
алг
нач
    цел N=30
    целтаб a[1:N]
    цел i, j, s
    нц для i от 1 до N
        ввод a[i]
    кц
    ...
кон
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.4) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

**С3** Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **добавить в кучу один камень** или **возвести количество камней в квадрат**. Например, имея кучу из 7 камней, за один ход можно получить кучу из 8 или 49 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится 100 или более. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 100 или больше камней.

В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 99$ .

Говорят, 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) При каких значениях числа  $S$  Петя может выиграть первым ходом?

Укажите все такие значения и выигрывающий ход Пети.

б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

2. Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причем (а) Петя не может выиграть первым ходом, но (б) Петя может выиграть своим вторым ходом, независимо от того, как будет ходить Ваня.

Для указанных значений  $S$  опишите выигрышную стратегию Пети.

3. Укажите такое значение  $S$ , при котором

– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и при этом

– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

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

**С4** По каналу связи передаётся последовательность положительных целых чисел  $X_1, X_2, \dots$  все числа не превышают 1000, их количество заранее неизвестно. Каждое число передаётся в виде отдельной текстовой строки, содержащей десятичную запись числа. Признаком конца передаваемой последовательности является число 0.

Участок последовательности от элемента  $X_T$  до элемента  $X_{T+N}$  называется подъёмом, если на этом участке каждое следующее число больше или равно предыдущему, причем участок нельзя расширить, т.е.

1)  $T = 1$  или  $X_{T-1} > X_T$

2)  $X_{T+N}$  – последний элемент последовательности или  $X_{T+N} > X_{T+N+1}$ .

Высотой подъёма называется разность  $X_{T+N} - X_T$ . Подъём считается значительным, если высота подъёма больше величины минимального элемента этого подъёма.

Напишите эффективную программу, которая вычисляет количество значительных подъёмов в заданной последовательности.

Программа должна вывести результаты в следующей форме:

Получено чисел: ...

Найдено значительных подъёмов: ...

Размер памяти, которую использует программа, не должен зависеть от длины переданной последовательности чисел.

Перед текстом программы кратко опишите алгоритм решения задачи.

*Пример входных данных:*

144  
17  
21  
27  
3  
7  
9  
11  
25  
0

*Пример выходных данных для приведённого выше примера входных данных:*

Получено чисел: 9

Найдено значительных подъёмов: 1

**Тренировочная работа № 4**

**по ИНФОРМАТИКЕ**

**17 мая 2013 года**

**11 класс**

**Вариант ИНФ1602**

**Район**

---

**Город (населённый пункт)**

---

**Школа**

---

**Класс**

---

**Фамилия**

---

**Имя**

---

**Отчество**

---

**Инструкция по выполнению работы**

На выполнение экзаменационной работы по информатике и ИКТ отводится 235 минут. Экзаменационная работа состоит из 3 частей, содержащих 32 задания. Рекомендуем не более 1,5 часов (90 минут) отвести на выполнение заданий частей 1 и 2, а остальное время – на часть 3

Часть 1 содержит 13 заданий (A1–A13). К каждому заданию даётся четыре варианта ответа, из которых только один правильный

Часть 2 состоит из 15 заданий с кратким ответом (B1–B15). К этим заданиям Вы должны самостоятельно сформулировать и записать ответ

Часть 3 состоит из 4 заданий (C1–C4). Для выполнения заданий этой части Вам необходимо написать развёрнутый ответ в произвольной форме

Все бланки ЕГЭ заполняются яркими чёрными чернилами. Допускается использование гелевой, капиллярной или перьевой ручек

При выполнении заданий Вы можете пользоваться черновиком. Обращаем Ваше внимание на то, что записи в черновике не будут учитываться при оценивании работы

Советуем выполнять задания в том порядке, в котором они даны. Для экономии времени пропускайте задание, которое не удаётся выполнить сразу, и переходите к следующему. Если после выполнения всей работы у Вас останется время, Вы сможете вернуться к пропущенным заданиям

Баллы, полученные Вами за выполненные задания, суммируются. Постарайтесь выполнить как можно больше заданий и набрать наибольшее количество баллов

*Желаем успеха!*

В экзаменационных заданиях используются следующие соглашения.

1. Обозначения для логических связок (операций):

а) *отрицание* (инверсия, логическое НЕ) обозначается  $\neg$  (например,  $\neg A$ );

б) *конъюнкция* (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );

в) *дизъюнкция* (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );

г) *импликация* (следование) обозначается  $\rightarrow$  (например,  $A \rightarrow B$ );

д) *тождество* обозначается  $\equiv$  (например,  $A \equiv B$ ); выражение  $A \equiv B$  истинно тогда и только тогда, когда значения  $A$  и  $B$  совпадают (либо они оба истинны, либо они оба ложны);

е) символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).

2. Два логических выражения, содержащие переменные, называются *равносильными* (эквивалентными), если значения этих выражений совпадают при любых значениях переменных. Так, выражения  $A \rightarrow B$  и  $(\neg A) \vee B$  равносильны, а  $A \vee B$  и  $A \wedge B$  – нет (значения выражений разные, например, при  $A = 1, B = 0$ ).

3. Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование). Таким образом,  $\neg A \wedge B \vee C \wedge D$  совпадает с  $((\neg A) \wedge B) \vee (C \wedge D)$ . Возможна запись  $A \wedge B \wedge C$  вместо  $(A \wedge B) \wedge C$ . То же относится и к дизъюнкции: возможна запись  $A \vee B \vee C$  вместо  $(A \vee B) \vee C$ .

4. Во всех задачах 1 Кбайт = 1024 байт, 1 Мбайт = 1024 Кбайт.

### Часть 1

**При выполнении заданий этой части в бланке ответов № 1 под номером выполняемого Вами задания (A1–A13) поставьте знак «X» в клеточке, номер которой соответствует номеру выбранного Вами ответа.**

**A1** Какое из приведённых выражений имеет наименьшее значение?

1)  $128_{10} + 64_{10} + 8_{10} + 4_{10}$

2)  $313_8$

3)  $E5_{16}$

4)  $11100111_2$

**A2** Между населёнными пунктами A, B, C, D, E, F, Z построены дороги, протяжённость которых приведена в таблице. (Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.)

	A	B	C	D	E	F	Z
A		4	10	15			34
B	4		9	6			
C	10	9		2			
D	15	6	2		3	11	15
E				3		8	13
F				11	8		4
Z	34			15	13	4	

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

1) 34

2) 30

3) 25

4) 22

**A3** Дан фрагмент таблицы истинности выражения F:

x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	F
0	1	1	0	1	1	0	1	1	1	1
1	0	1	1	0	0	1	1	1	0	0
1	0	0	0	1	1	0	0	1	0	1

Каким выражением может быть F?

1)  $(x1 \vee \neg x2) \wedge (x3 \vee \neg x4) \wedge x5 \wedge \neg x6 \wedge x7 \wedge x8 \wedge \neg x9 \wedge x10$

2)  $(x1 \wedge \neg x2) \vee (x3 \wedge \neg x4) \vee \neg x5 \vee \neg x6 \vee x7 \vee x8 \vee \neg x9 \vee x10$

3)  $(\neg x1 \wedge x2) \vee (\neg x3 \wedge x4) \vee x5 \vee x6 \vee \neg x7 \vee \neg x8 \vee \neg x9 \vee x10$

4)  $(\neg x1 \vee x2) \wedge (\neg x3 \vee x4 \wedge \neg x5 \wedge x6 \wedge \neg x7 \wedge \neg x8 \wedge x9 \wedge \neg x10)$

**A4** 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 находится 6 файлов:

inmar.xls  
komarik.xlsx  
komarik.xxx  
marka.xlsx  
romario.xls  
smart.xlsx

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

inmar.xls  
komarik.xlsx  
romario.xls  
smart.xlsx

1) \*?mar\*.x\*

2) ?mar\*.xls\*

3) \*?mar\*.xls\*

4) \*mar?\* .xls\*

**A5** Автомат получает на вход трёхзначное десятичное число, в котором все цифры нечётные. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также вторая и третья цифры.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

**Пример.** Исходное число: 175. Суммы:  $1+7 = 8$ ;  $7+5 = 12$ ; Результат: 128.

Определите, какое из перечисленных ниже чисел может быть результатом работы автомата.

1) 148

2) 167

3) 178

4) 200

**A6** Во фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите, сколько всего родных братьев и сестёр есть у Жук М.Б.

ID	Фамилия И.О.	Пол
1674	Жук М.Б.	Ж
1702	Баль А.П.	М
1769	Черняк И.Б.	М
1834	Рерих А.И.	Ж
2046	Черняк П.И.	М
2060	Радек П.А.	Ж
2094	Черняк Б.Ф.	М
2192	Чиж Д.К.	Ж
2425	Рерих Л.А.	Ж
2435	Черняк А.Б.	Ж
2607	Малеев К.Г.	М
2679	Баль П.А.	М
2816	Черняк Т.И.	Ж
2946	Панина Р.Г.	Ж
2968	Тесленко Г.Р.	Ж
2997	Рерих В.И.	Ж
...	...	...

ID Родителя	ID Ребёнка
1702	2679
1769	2046
1769	2816
1769	2997
2094	1674
2094	1769
2094	2435
2192	1674
2192	1769
2192	2435
2435	2679
2968	2997
2968	2046
2968	2816
2997	2060
2997	2425
...	...

1) 1

2) 2

3) 3

4) 0

**A7** В ячейке E15 электронной таблицы записана формула. Эту формулу скопировали в ячейки D17 и C18. В соответствии с формулой, полученной в ячейке D17, значение в этой ячейке равно разности значений в ячейках D32 и C32; в соответствии с формулой, полученной в ячейке C18, значение в этой ячейке равно разности значений в ячейках D33 и B32.

Укажите, какая формула могла быть написана в ячейке E15.

*Примечание: знак \$ используется для обозначения абсолютной адресации.*

1) =E\$32 – \$D30

2) =\$D\$32 – \$B\$32

3) =\$D\$31 – \$C\$32

4) =\$D30 – D\$32

**A8** В течение одной минуты производилась четырехканальная (квадро) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. Сжатие данных не производилось. Какая из приведённых ниже величин наиболее близка к размеру полученного файла?

1) 20 Мбайт

2) 30 Мбайт

3) 40 Мбайт

4) 60 Мбайт



**A9** По каналу связи передаются сообщения, содержащие только 4 буквы: А, И, С, Т.

В любом сообщении больше всего букв А, следующая по частоте буква – С, затем – И. Буква Т встречается реже, чем любая другая.

Для передачи сообщений нужно использовать неравномерный двоичный код, допускающий однозначное декодирование; при этом сообщения должны быть как можно короче. Шифровальщик может использовать один из перечисленных ниже кодов. Какой код ему следует выбрать?

- 1) А – 0, И – 1, С – 00, Т – 11      2) С – 1, И – 0, А – 01, Т – 10  
3) А – 1, И – 01, С – 001, Т – 000      4) С – 0, И – 11, А – 101, Т – 100

**A10** На числовой прямой даны два отрезка:  $P = [40, 60]$  и  $Q = [20, 90]$ . Выберите такой отрезок А, чтобы формула

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

была тождественно истинна, то есть принимала значение 1 при любом значении переменной  $x$ . Если таких отрезков несколько, укажите тот, который имеет меньшую длину.

- 1) [17, 43]      2) [17, 73]      3) [37, 53]      4) [37, 63]

**A11** При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 10 символов и содержащий только символы из 7-буквенного набора Н, О, Р, С, Т, У, Х. В базе данных для хранения сведений о каждом пользователе отведено одинаковое целое число байт, при этом для хранения сведений о 100 пользователях используется 1500 байт. Для каждого пользователя хранятся пароль и дополнительные сведения. Для хранения паролей используют посимвольное кодирование, все символы кодируются одинаковым и минимально возможным количеством бит. Сколько бит отведено для хранения дополнительных сведений о каждом пользователе?

- 1) 88      2) 90      3) 91      4) 96

**A12** Ниже представлен записанный на разных языках программирования фрагмент одной и той же программы. В программе описан одномерный целочисленный массив А, в представленном фрагменте программы обрабатываются элементы массива с индексами от 1 до 10.

<b>Бейсик</b>	n = 10 FOR i = 1 TO n A(n+1-i) = 2*A(i) NEXT i
<b>Паскаль</b>	n := 10; for i := 1 to n do begin A[n+1-i] := 2*A[i]; end;
<b>Си</b>	n = 10; for (i = 1; i <= n; i++) A[n+1-i] = 2*A[i];
<b>Алгоритмический язык</b>	n := 10 <u>нц</u> <u>для</u> i <u>от</u> 1 <u>до</u> n A[n+1-i] := 2*A[i] <u>кц</u>

Перед началом выполнения фрагмента элементы массива имеют значения соответственно

$$1, 2, 4, 8, 16, 32, 64, 128, 256, 512, \text{ т.е. } A[k] = 2^{k-1}, k = 1, \dots, 10.$$

Укажите значение, которое после выполнения указанного фрагмента программы имеют два или более рассмотренных в этом фрагменте элемента массива. Если таких чисел несколько, укажите наибольшее из них.

- 1) такого значения нет      2) 8  
3) 32      4) 128

**A13** Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости, состоит из 8 команд. Четыре команды – это команды-приказы:

<b>вверх</b>	<b>вниз</b>	<b>влево</b>	<b>вправо</b>
--------------	-------------	--------------	---------------

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:

<b>сверху свободно</b>	<b>снизу свободно</b>	<b>слева свободно</b>	<b>справа свободно</b>
------------------------	-----------------------	-----------------------	------------------------

Цикл

ПОКА *условие*

*последовательность команд*

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ *условие*

ТО *команда1*

ИНАЧЕ *команда2*

КОНЕЦ ЕСЛИ

выполняется *команда1* (если условие истинно) или *команда2* (если условие ложно).

В конструкциях ПОКА и ЕСЛИ *условие* может содержать команды проверки, а также слова И, ИЛИ, НЕ, обозначающие логические операции.

Если РОБОТ начнёт движение в сторону находящейся рядом с ним стены, то он разрушится, и программа прервётся.

Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка F6)?

НАЧАЛО

ПОКА **снизу свободно** ИЛИ **справа свободно**

ЕСЛИ **справа свободно**

ТО

**вправо**

КОНЕЦ ЕСЛИ

ЕСЛИ **снизу свободно**

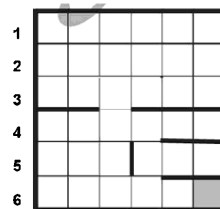
ТО

**вниз**

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ



A B C D E F

1) 9

2) 11

3) 15

4) 18

## Часть 2

*Ответом к заданиям этой части (B1–B15) является число, последовательность букв или цифр. Впишите ответы сначала в текст работы, а затем перенесите их в бланк ответов № 1 справа от номера соответствующего задания, начиная с первой клеточки, без пробелов, запятых и других дополнительных символов. Каждую букву или цифру пишите в отдельной клеточке в соответствии с приведёнными в бланке образцами.*

**B1** У исполнителя Аккорд две команды, которым присвоены номера:

1. отними 1

2. умножь на 5

Выполняя первую из них, Аккорд отнимает от числа на экране 1, а выполняя вторую, умножает это число на 5.

Запишите порядок команд в программе, которая содержит не более 5 команд и переводит число 1 в число 99.

В ответе указывайте лишь номера команд, пробелы между цифрами не ставьте. Так, для программы

**умножь на 5**

**отними 1**

**отними 1**

нужно написать: 211. Эта программа преобразует, например, число 5 в число 23.

Ответ:

**В2** Определите значение переменной **c** после выполнения следующего фрагмента программы (записанного ниже на разных языках программирования). Ответ запишите в виде целого числа.

<b>Бейсик</b>	<pre>a = 50 b = 12 a = a/5 - b/3 IF 4*a &gt; b THEN   c = a - 2*b + 37 ELSE   c = a + 2*b + 37 END IF</pre>
<b>Паскаль</b>	<pre>a := 50; b := 12; a := a/5 - b/3; if 4*a &gt; b then   c := a - 2*b + 37 else   c := a + 2*b + 37;</pre>
<b>Си</b>	<pre>a = 50; b = 12; a = a/5 - b/3; if (4*a &gt; b)   c = a - 2*b + 37; else   c = a + 2*b + 37;</pre>
<b>Алгоритмический</b>	<pre>a := 50 b := 12 a := a/5 - b/3 если 4*a &gt; b   то c := a - 2*b + 37   иначе c := a + 2*b + 37 все</pre>

Ответ:

**В3** Дан фрагмент электронной таблицы.

	A	B	C
<b>1</b>	20		35
<b>2</b>	=C1-2*B1*B1	=(B1*B1*B1-4)/A1	=C1-8*B1

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



Известно, что все значения диапазона **A1:C2** имеют один и тот же знак.

Ответ:

**В4** На световой панели в ряд расположены 8 лампочек. Каждая из первых двух лампочек может гореть красным, жёлтым или зелёным цветом. Каждая из остальных шести лампочек может гореть одним из двух цветов – красным или белым. Сколько различных сигналов можно передать с помощью панели (все лампочки должны гореть, порядок цветов имеет значение)?

Ответ:

**В5** Определите, что будет напечатано в результате выполнения программы (записанной ниже на разных языках программирования):

<b>Бейсик</b>	<pre>DIM N, S AS INTEGER N = 50 S = 1 WHILE S &lt; 1000   S = S * 2   N = N + 10 WEND PRINT N</pre>
---------------	---

<b>Паскаль</b>	<pre>var n, s: integer; begin   n := 50;   s := 1;   while s &lt; 1000 do   begin     s := s * 2;     n := n + 10;   end;   write(n); end.</pre>
<b>Си</b>	<pre>#include&lt;stdio.h&gt; void main() {   int n, s;   n = 50;   s = 1;   while (s &lt; 1000)   {     s = s * 2;     n = n + 10;   }   printf("%d", n); }</pre>
<b>Алгоритмический</b>	<pre><u>алг</u> <u>нач</u>   <u>цел</u> n, s   n := 50   s := 1   <u>нц пока</u> s &lt; 1000     s := s * 2     n := n + 10   <u>кц</u>   <u>вывод</u> n <u>кон</u></pre>

Ответ:

**В6** Алгоритм вычисления значения функции  $F(n)$ , где  $n$  – натуральное число, задан следующими соотношениями:

$F(1) = 1;$

$F(n) = 2 \cdot F(n-1) + 1$  при  $n > 1$ .

Чему равно значение функции  $F(6)$ ?

*В ответе запишите только натуральное число.*

Ответ:

**В7** Запись числа  $N$  в системе счисления с основанием 7 содержит две цифры, запись этого числа в системе счисления с основанием 6 содержит три цифры, а запись в системе счисления с основанием 11 заканчивается на 2. Чему равно  $N$ ?

Ответ:

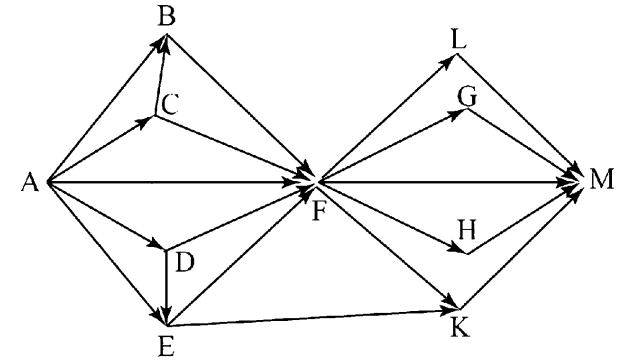
**В8** Ниже на четырёх языках записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа  $a$  и  $b$ . Укажите наибольшее из таких чисел  $x$ , при вводе которых алгоритм печатает сначала 2, а потом 17.

<b>Бейсик</b>	<pre>DIM X, A, B AS INTEGER INPUT X A = 0: B = 0 WHILE X &gt; 0   A = A+1   B = B + (X MOD 100)   X = X\100 WEND PRINT A PRINT B</pre>
<b>Паскаль</b>	<pre>var x, a, b: integer; begin   readln(x);   a := 0; b := 0;   while x &gt; 0 do   begin     a := a+1;     b := b + (x mod 100);     x := x div 100;   end;   writeln(a); write(b); end.</pre>

<b>Си</b>	<pre>#include &lt;stdio.h&gt; void main() {     int x, a, b;     scanf("%d", &amp;x);     a = 0; b = 0;     while (x &gt; 0) {         a = a+1;         b = b + (x%100);         x = x/100;     }     printf("%d\n%d", a, b); }</pre>
<b>Алгоритмический</b>	<pre><u>алг</u> <u>нач</u>     <u>цел</u> x, a, b     <u>ввод</u> x     a:=0; b:=0     <u>нц пока</u> x &gt; 0         a := a+1         b := b+mod(x,100)         x := div(x,100)     <u>кц</u>     <u>вывод</u> a, <u>нс</u>, b <u>кон</u></pre>

Ответ:

- В9** На рисунке – схема дорог, связывающих города A, B, C, D, E, F, G, H, K, L, M. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города A в город M?



Ответ:

- В10** Документ (без упаковки) можно передать по каналу связи с одного компьютера на другой за 1 мин. 20 с. Если предварительно упаковать документ архиватором, передать упакованный документ, а потом распаковать на компьютере получателя, то общее время передачи (включая упаковку и распаковку) составит 20 с. При этом на упаковку и распаковку данных всего ушло 10 с. Размер исходного документа 24 Мбайт. Чему равен размер упакованного документа (в Мбайт)? В ответе запишите только число.

Ответ:

**В11** 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 заданным IP-адресу узла и маске.

По заданным IP-адресу узла и маске определите адрес сети.

IP-адрес узла: 248.137.249.32

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 без использования точек.

A	B	C	D	E	F	G	H
255	249	248	224	137	32	8	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.

Ответ:

**В12** В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» – символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тыс.)
Ильф & Петров & Остап	800
Ильф & Петров & Бендер	600
Ильф & Петров & Остап & Бендер	500

Какое количество страниц (в тыс.) будет найдено по запросу

$(Ильф \& Петров \& Остап) | (Ильф \& Петров \& Бендер)?$

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ:

**В13** У исполнителя Полтора две команды, которым присвоены номера:

1. прибавь один,

2. умножь на полтора.

Первая из них увеличивает на 1 число на экране, вторая увеличивает это число в 1,5 раза, если число чётное. К нечётным числам вторая команда неприменима.

Программа для Полтора – это последовательность команд.

Сколько существует программ, которые число 1 преобразуют в число 22?

Ответ:

**В14** Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках):

<b>Бейсик</b>	<pre> DIM A, B, T, M, R AS INTEGER A = -7: B = 25 M = A: R = F(A) FOR T = A TO B+7     IF F(T) &lt; R THEN         M = T         R = F(T)     END IF NEXT T PRINT M FUNCTION F(x)     F = 182 - 2*(19+x)*(19+x) END FUNCTION </pre>
---------------	---

<b>Паскаль</b>	<pre> var a,b,t,M,R :integer; Function F(x: integer):integer; begin   F := 182 - 2*(19+x)*(19+x); end; BEGIN a := -7; b := 25; M := a; R := F(a); for t := a to b+7 do begin   if (F(t) &lt; R) then begin     M := t;     R := F(t);   end; end; write(M); END.</pre>
----------------	--

<b>Си</b>	<pre> #include&lt;stdio.h&gt; int F(int x) {   return 182 - 2*(19+x)*(19+x); } void main() {   int a, b, t, M, R;   a = -7; b = 25;   M = a; R = F(a);   for (t=a; t&lt;=b+7; t++){     if (F(t) &lt; R) {       M = t; R = F(t);     }   }   printf("%d", M); }</pre>
-----------	--

<b>Алгоритмический</b>	<pre> алг нач   цел a, b, t, R, M   a := -7; b := 25   M := a; R := F(a)   нц для t от a до b+7     если F(t) &lt; R       то         M := t; R := F(t)     все   кц   вывод M кон алг цел F(цел x) нач   знач := 182 - 2*(19+x)*(19+x) кон</pre>
------------------------	---

Ответ:

**B15** Сколько существует различных наборов значений логических переменных  $x_1, x_2, x_3, x_4, x_5, x_6, y_1, y_2, y_3, y_4, y_5, y_6$ , которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \rightarrow x_2) \wedge (x_2 \rightarrow x_3) \wedge (x_3 \rightarrow x_4) \wedge (x_4 \rightarrow x_5) \wedge (x_5 \rightarrow x_6) = 1$$

$$(y_2 \rightarrow y_1) \wedge (y_3 \rightarrow y_2) \wedge (y_4 \rightarrow y_3) \wedge (y_5 \rightarrow y_4) \wedge (y_6 \rightarrow y_5) = 1$$

$$x_1 \vee y_1 = 1$$

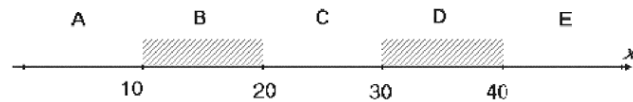
В ответе не нужно перечислять все различные наборы значений переменных  $x_1, x_2, x_3, x_4, x_5, x_6, y_1, y_2, y_3, y_4, y_5, y_6$ , при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

Ответ:

## Часть 3

Для записи ответов к заданиям этой части (C1–C4) используйте бланк ответов № 2. Запишите сначала номер задания (C1 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

**C1** Требовалось написать программу, при выполнении которой с клавиатуры считывается координата точки на прямой ( $x$  – действительное число) и определяется принадлежность этой точки заданной области (включая границы). Программист торопился и написал программу неправильно.

**Бейсик**

```
INPUT x
IF x<=30 THEN
  IF x<=20 THEN
    IF x<=10 THEN
      PRINT "не принадлежит"
    ELSE
      PRINT "принадлежит"
    END IF
  END IF
END IF
END IF
END
```

**Паскаль**

```
var x: real;
begin
  readln(x);
  if x<=30 then
    if x<=20 then
      if x<=10 then
        write('не принадлежит')
      else
        write('принадлежит')
      end.
    end.
  end.
```

**Си**

```
#include <stdio.h>
void main()
{
  float x;
  scanf("%f",&x);
  if (x<=30)
    if (x<=20)
      if (x<=10)
        printf("не принадлежит");
      else
        printf("принадлежит");
    }
}
```

**Алгоритмический язык**алгнач

```
вещ x
ввод x
если x<=30 то
  если x<=20 то
    если x<=10 то
      вывод 'не принадлежит'
    иначе
      вывод 'принадлежит'
  все
все
все
кон
```

Последовательно выполните следующее.

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (A, B, C, D и E). Точки, лежащие на границах областей (то есть, точки 10, 20, 30 и 40) при заполнении таблицы не рассматриваются.

Область	Условие 1 ( $x \leq 30$ )	Условие 2 ( $x \leq 20$ )	Условие 3 ( $x \leq 10$ )	Программа выведет	Область обрабатывается верно
A					
B					
C					
D					
E					



В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.», если программа ведет себя по-разному для разных значений, принадлежащих данной области. В столбце «Программа выведет» укажите, что программа выведет на экран. Если программа ничего не выводит, поставьте «—» (прочерк). Если для разных значений, принадлежащих области, будут выведены разные тексты, напишите «не изв.». 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.)

**C2** Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Элемент массива называется хорошим, если это двузначное число, причём цифра в разряде десятков больше, чем цифра в разряде единиц. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести сумму всех хороших элементов массива.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

#### Паскаль

```
const
    N=30;
var
    a: array [1..N] of integer;
    i, j, s: integer;
begin
    for i:=1 to N do
        readln(a[i]);
    ...
end.
```

#### Бейсик

```
N=30
DIM A(N) AS INTEGER
DIM I, J, S AS INTEGER
FOR I = 1 TO N
    INPUT A(I)
NEXT I
...
END
```

#### Си

```
#include <stdio.h>
#define N 30
void main(){
    int a[N];
    int i, j, s;
    for (i=0; i<N; i++)
        scanf("%d", &a[i]);
    ...
}
```

#### Алгоритмический язык

алг

нач

```
цел N=30
целтаб a[1:N]
цел i, j, s
нц для i от 1 до N
    ввод a[i]
```

кц

...

кон

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.4) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

**С3** Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **добавить в кучу один камень** или **возвести количество камней в квадрат**. Например, имея кучу из 7 камней, за один ход можно получить кучу из 8 или 49 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится 100 или более. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 100 или больше камней.

В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 99$ .

Говорят, 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) При каких значениях числа  $S$  Петя может выиграть первым ходом? Укажите все такие значения и выигрывающий ход Пети.

б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

2. Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причем (а) Петя не может выиграть первым ходом, но (б) Петя может выиграть своим вторым ходом, независимо от того, как будет ходить Ваня.

Для указанных значений  $S$  опишите выигрышную стратегию Пети.

3. Укажите такое значение  $S$ , при котором

– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и при этом

– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

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

**С4** По каналу связи передаётся последовательность положительных целых чисел  $X_1, X_2, \dots$  все числа не превышают 1000, их количество заранее неизвестно. Каждое число передаётся в виде отдельной текстовой строки, содержащей десятичную запись числа. Признаком конца передаваемой последовательности является число 0.

Участок последовательности от элемента  $X_T$  до элемента  $X_{T+N}$  называется подъёмом, если на этом участке каждое следующее число больше или равно предыдущему, причем участок нельзя расширить, т.е.

1)  $T = 1$  или  $X_{T-1} > X_T$

2)  $X_{T+N}$  – последний элемент последовательности или  $X_{T+N} > X_{T+N+1}$ .

Высотой подъёма называется разность  $X_{T+N} - X_T$ . Подъём считается значительным, если высота подъёма больше величины минимального элемента этого подъёма.

Напишите эффективную программу, которая вычисляет количество значительных подъёмов в заданной последовательности.

Программа должна вывести результаты в следующей форме:

Получено чисел: ...

Найдено значительных подъёмов: ...

Размер памяти, которую использует программа, не должен зависеть от длины переданной последовательности чисел.

Перед текстом программы кратко опишите алгоритм решения задачи.

*Пример входных данных:*

144  
17  
21  
27  
3  
7  
9  
11  
25  
0

*Пример выходных данных для приведённого выше примера входных данных:*

Получено чисел: 9

Найдено значительных подъёмов: 1

**Ответы к заданиям с выбором ответа**

№ задания	Ответ
A1	4
A2	4
A3	2
A4	4
A5	3
A6	1
A7	1

№ задания	Ответ
A8	2
A9	4
A10	2
A11	3
A12	4
A13	1

**Ответы к заданиям с кратким ответом**

№ задания	Ответ
B1	12211
B2	5
B3	3
B4	288
B5	120
B6	31
B7	34
B8	1300

№ задания	Ответ
B9	36
B10	4
B11	DFDH
B12	1400
B13	32
B14	28
B15	31

**Ответы к заданиям с выбором ответа**

№ задания	Ответ
A1	2
A2	3
A3	3
A4	3
A5	1
A6	2
A7	4

№ задания	Ответ
A8	2
A9	3
A10	4
A11	2
A12	3
A13	2

**Ответы к заданиям с кратким ответом**

№ задания	Ответ
B1	21221
B2	19
B3	4
B4	576
B5	150
B6	63
B7	46
B8	1700

№ задания	Ответ
B9	37
B10	3
B11	СЕСН
B12	900
B13	44
B14	32
B15	43

**Ответы к заданиям с выбором ответа**

№ задания	Ответ
A1	4
A2	3
A3	2
A4	3
A5	3
A6	2
A7	1

№ задания	Ответ
A8	2
A9	4
A10	4
A11	3
A12	3
A13	1

**Ответы к заданиям с кратким ответом**

№ задания	Ответ
B1	21221
B2	5
B3	4
B4	288
B5	150
B6	31
B7	46
B8	1300

№ задания	Ответ
B9	37
B10	4
B11	СЕСН
B12	1400
B13	44
B14	28
B15	43

**Ответы к заданиям с выбором ответа**

№ задания	Ответ
A1	2
A2	4
A3	3
A4	4
A5	1
A6	1
A7	4

№ задания	Ответ
A8	2
A9	3
A10	2
A11	2
A12	4
A13	2

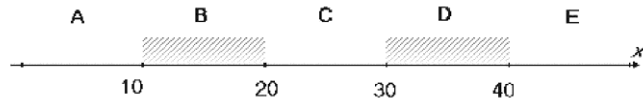
**Ответы к заданиям с кратким ответом**

№ задания	Ответ
B1	12211
B2	19
B3	3
B4	576
B5	120
B6	63
B7	34
B8	1700

№ задания	Ответ
B9	37
B10	3
B11	DFDH
B12	900
B13	32
B14	32
B15	31

**Критерии оценивания заданий с развёрнутым ответом**

**C1** Требовалось написать программу, при выполнении которой с клавиатуры считывается координата точки на прямой ( $x$  – действительное число) и определяется принадлежность этой точки заданной области (включая границы). Программист торопился и написал программу неправильно.

**Бейсик**

```
INPUT x
IF x<=30 THEN
  IF x<=20 THEN
    IF x<=10 THEN
      PRINT "не принадлежит"
    ELSE
      PRINT "принадлежит"
    END IF
  END IF
END IF
END IF
END
```

**Паскаль**

```
var x: real;
begin
  readln(x);
  if x<=30 then
    if x<=20 then
      if x<=10 then
        write('не принадлежит')
      else
        write('принадлежит')
      end if
    end if
  end if
end.
```

**Си**

```
#include <stdio.h>
void main()
{
  float x;
  scanf("%f",&x);
  if (x<=30)
    if (x<=20)
      if (x<=10)
        printf("не принадлежит");
      else
        printf("принадлежит");
  }
}
```

**Алгоритмический язык**

```
алг
нач
  вещ x
  ввод x
  если x<=30 то
    если x<=20 то
      если x<=10 то
        вывод 'не принадлежит'
      иначе
        вывод 'принадлежит'
      все
    все
  все
кон
```

Последовательно выполните следующее.

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (A, B, C, D и E). Точки, лежащие на границах областей (то есть, точки 10, 20, 30 и 40) при заполнении таблицы не рассматриваются.

Область	Условие 1 ( $x \leq 30$ )	Условие 2 ( $x \leq 20$ )	Условие 3 ( $x \leq 10$ )	Программа выведет	Область обрабатывается верно
A					
B					
C					
D					
E					

В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.», если программа ведет себя по-разному для разных значений, принадлежащих данной области. В столбце «Программа выведет» укажите, что программа выведет на экран. Если программа ничего не выводит, поставьте «—» (прочерк). Если для разных значений, принадлежащих области, будут выведены разные тексты, напишите «не изв.». 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.)

**Содержание верного ответа и указания по оцениванию**

(допускаются иные формулировки ответа, не искажающие его смысла)

Элементы ответа:

1.

Область	Условие 1 ( $x \leq 30$ )	Условие 2 ( $x \leq 20$ )	Условие 3 ( $x \leq 10$ )	Программа выведет	Область обрабатывается верно
A	да	да	да	не принадлежит	Да
B	да	да	нет	принадлежит	Да
C	да	нет	—	—	Нет
D	нет	—	—	—	Нет
E	нет	—	—	—	Нет

2. Возможная доработка (Паскаль):

```
if (x>=10) and (x<=20) or (x>=30) and (x<=40) then
  write('принадлежит')
else
  write('не принадлежит')
```

Возможны и другие способы доработки.

Например:

```
if x>=10 then
  if x<=20 then
    write('принадлежит')
  else
    if x>=30 then
      if x<=40 then
        write('принадлежит')
      else
        write('не принадлежит')
    else
      write('не принадлежит')
else
  write('не принадлежит')
```

Другой пример:

```
if abs(abs(x-25)-10)<=5 then
  write('принадлежит')
else
  write('не принадлежит')
```

**Обратите внимание!** В задаче требовалось выполнить три действия: указать для каждой области, как будет работать программа, что она выведет на экран и правильно ли это (в виде таблицы) и исправить две ошибки.

Баллы за данное задание начисляются как сумма баллов за верное выполнение каждого действия.

1. Верное заполнение предложенной таблицы.

2. Неправильное использование условного оператора, в результате чего при невыполнении первого или второго условия программа не выдавала ничего

(отсутствуют случаи ELSE). Исправлением этой ошибки может быть либо добавление случая ELSE к каждому условию IF, либо объединение всех условий IF в одно при помощи конъюнкции.

В сложных случаях это действие считается выполненным, если программа выдаёт одно из двух сообщений «принадлежит» или «не принадлежит» для любых чисел  $x$ , при этом программа не стала работать хуже, чем раньше, то есть для всех точек, для которых программа ранее выдавала верный ответ, доработанная программа также должна выдавать верный ответ.

3. Приведённых трёх ограничений недостаточно для описания двух областей (потеряно условие  $x \leq 40$ ). Кроме того, необходимо учесть, что области не соединены. Исправлением этой ошибки может быть разбиение области на две части и использование дизъюнкции, либо использование сложной (для выведения) математической конструкции ( $||x-25|-10| \leq 5$ ), либо использование сложной комбинации каскадных условий.

В сложных случаях это действие считается выполненным, если верно определены заштрихованные области, то есть программа выводит сообщение «принадлежит» для всех точек закрашенных областей, и только для них, для точек вне заштрихованных областей программа выводит «не принадлежит» или не выводит ничего.

Указания по оцениванию	Баллы
Правильно выполнены оба пункта задания. Исправлены две ошибки. Программа для всех чисел $x$ верно определяет принадлежность точки заштрихованной области. В работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения	3
1. Правильно выполнены два действия из трёх (исправлены обе ошибки, но в первом пункте задания не приведена таблица (либо таблица содержит ошибки более чем в одной строке), либо приведена таблица (которая содержит ошибки не более чем в одной строке), но исправлена только одна ошибка программы). При написании операций сравнения допускается одно неправильное использование строгих/нестрогих неравенств (считается несущественной ошибкой, погрешностью записи). Например, вместо « $x \leq 40$ » используется « $x < 40$ ». 2. Или выполнены все три действия, но при этом в логическом выражении неверно учтены приоритеты логических операций (не расставлены или неправильно расставлены скобки в выражениях)	2
Правильно выполнено только одно действие из трёх, то есть либо только приведена таблица, которая содержит ошибки в не более чем двух строках, либо таблица не приведена (или приведена и содержит ошибки более чем в двух строках), но исправлена одна ошибка программы. При оценивании этого задания на 1 балл допускается не учитывать корректность работы программ на точках границ областей (вместо нестрогих неравенств в решении были использованы строгие неравенства или наоборот)	1
Все пункты задания выполнены неверно (таблица анализа правильности алгоритма не приведена либо содержит три и более строк с ошибками, программа не приведена либо ни одна из двух ошибок не исправлена)	0
<i>Максимальный балл</i>	
	3

**C2** Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Элемент массива называется хорошим, если это двузначное число, причём цифра в разряде десятков больше, чем цифра в разряде единиц. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести сумму всех хороших элементов массива.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

#### Паскаль

```
const
  N=30;
var
  a: array [1..N] of integer;
  i, j, s: integer;
begin
  for i:=1 to N do
    readln(a[i]);
  ...
end.
```

#### Бейсик

```
N=30
DIM A(N) AS INTEGER
DIM I, J, S AS INTEGER
FOR I = 1 TO N
  INPUT A(I)
NEXT I
...
END
```

#### Си

```
#include <stdio.h>
#define N 30
void main(){
  int a[N];
  int i, j, s;
  for (i=0; i<N; i++)
    scanf("%d", &a[i]);
  ...
}
```

#### Алгоритмический язык

```
алг
нач
  цел N=30
  целтаб a[1:N]
  цел i, j, s
  нц для i от 1 до N
    ввод a[i]
  кц
  ...
кон
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.4) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

---

**Содержание верного ответа и указания по оцениванию**  
(допускаются иные формулировки ответа, не искажающие его смысла)

#### Пример программы на языке Паскаль

```
s:=0;
for i:=1 to N do begin
  if (10<=a[i]) and (a[i]<=99) and
    ((a[i] div 10) > (a[i] mod 10) ) then s:=s+a[i];
end;
writeln(s);
```

#### Пример программы на языке Бейсик

```
S = 0
FOR I = 1 TO N
  IF 10 <=A(I) AND A(I)<=99 AND A(I)\10 > A(I) MOD 10 THEN
    S = S + A(I)
  END IF
NEXT I
PRINT S
```

#### Пример программы на языке Си

```
s=0;
for (i=0; i<N; i++) {
  if (10<=a[i] && a[i]<=99 && a[i]/10 > a[i]%10)
    s = s + a[i];
}
printf("%d", s);
```

**Пример программы на алгоритмическом языке**

```

s:=0
нц для i от 1 до N
  если 10<=a[i]<=99 и div(a[i],10) > mod(a[i],10)
    то s:=s+a[i]
  все
кц
вывод s

```

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Допускается запись алгоритма на другом языке, использующая аналогичные переменные. В случае, если язык программирования использует типизированные переменные, описания переменных должны быть аналогичны описаниям переменных в тексте задания. Использование нетипизированных или необъявленных переменных возможно только в случае, если это допускается языком программирования, при этом количество переменных и их идентификаторы должны соответствовать условию задачи. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы	2
В любом варианте решения может присутствовать не более одной ошибки из числа следующих: 1) Не инициализируется или неверно инициализируется переменная S. 2) Неверно осуществляется проверка того, что элемент массива – хороший. 3) Вместо проверки того, что элемент хороший, аналогичная проверка выполняется для индекса элемента. 4) Неверно осуществляется накопление суммы в цикле (например, s:=a[i];). 5) Отсутствует вывод ответа. 6) Используется переменная, не объявленная в разделе описания переменных. 7) Не указано или неверно указано условие завершения цикла. 8) Индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно. 9) Неверно расставлены операторные скобки	1
Ошибок, перечисленных в п. 1–9, две или больше, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

**С3** Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **добавить в кучу один камень** или **возвести количество камней в квадрат**. Например, имея кучу из 7 камней, за один ход можно получить кучу из 8 или 49 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится 100 или более. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 100 или больше камней.

В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 99$ .

Говорят, 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) При каких значениях числа  $S$  Петя может выиграть первым ходом? Укажите все такие значения и выигрывающий ход Пети.

б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

2. Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причем (а) Петя не может выиграть первым ходом, но (б) Петя может выиграть своим вторым ходом, независимо от того, как будет ходить Ваня.

Для указанных значений  $S$  опишите выигрышную стратегию Пети.

3. Укажите такое значение  $S$ , при котором  
– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и при этом  
– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

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

**Содержание верного ответа и указания по оцениванию**

(допускаются иные формулировки ответа, не искажающие его смысла)

1. а) Петя может выиграть, если  $S = 10, \dots, 99$ . Пете достаточно возвести количество камней в квадрат. При  $S < 10$  получить за один ход 100 или больше камней невозможно.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет  $S = 9$  камней. Тогда после первого хода Пети в куче будет 10 камней или 81 камень. В обоих случаях Ваня возводит количество камней в квадрат и выигрывает в один ход.

2. Возможные значения  $S$ : 3, 8. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 9 камней (при  $S=3$  он возводит количество камней в квадрат; при  $S=8$  – добавляет 1 камень). Эта позиция разобрана в п. 1 б. В ней игрок, который будет ходить (теперь это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.



3. Возможное значение  $S$ : 7. После первого хода Пети в куче будет 8 или 49 камней. Если в куче станет 49 камней, Ваня возводит количество камней в квадрат и выигрывает своим первым ходом. Ситуация, когда в куче 8 камней разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Ваня) выигрывает своим вторым ходом.

В таблице изображено дерево возможных партий при описанной стратегии Вани. Заключительные позиции (в них выигрывает Ваня) подчеркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

Исх. положение	Положения после очередных ходов			
	1-й ход Пети (разобраны все ходы)	1-й ход Вани (только ход по стратегии)	2-й ход Пети (разобраны все ходы)	2-й ход Вани (только ход по стратегии)
7	$7+1=8$	$8+1=9$	$9+1=10$	$10*10=100$
	$7*7=49$	$49*49=2401$	$9*9=81$	$81*81=6561$

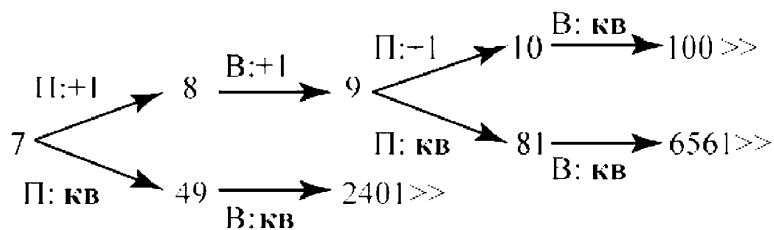


Рис. 1. Дерево всех партий, возможных при Ваниной стратегии. Знаком >> обозначены позиции, в которых партия заканчивается

В задаче от ученика требуется выполнить 3 задания. Их трудность возрастает. Количество баллов в целом соответствует количеству выполненных заданий (подробнее см. ниже).

Ошибка в решении, не искажающая основного замысла и не приведшая к неверному ответу, например, арифметическая ошибка при вычислении количества камней в заключительной позиции, при оценке решения не учитывается.

Первое задание считается выполненным полностью, если выполнены полностью оба пункта а) и б). Пункт а) считается выполненным полностью, если правильно указаны все позиции, в которых Петя выигрывает первым ходом, и указано, каким должен быть первый ход. Пункт б) считается выполненным, если правильно указана позиция, в которой Ваня выигрывает первым ходом, и описана стратегия Вани, т. е. показано, как Ваня может получить кучу, в которой содержится нужное количество камней, при любом ходе Пети.

Первое задание считается выполненным частично, если (а) правильно указаны все позиции, в которых Петя выигрывает первым ходом, (б) правильно указана позиция, в которой Ваня выигрывает первым ходом, и явно сказано, что при любом ходе Пети Ваня может получить кучу, которая содержит нужное для выигрыша количество камней. Отличие от полного решения в том, что явно не указаны ходы, ведущие к выигрышу.

Второе задание выполнено, если правильно указаны обе позиции, выигрышные для Пети, и описана соответствующая стратегия Пети – так, как это написано в примере решения, или другим способом, например, с помощью дерева всех партий, возможных при выбранной стратегии Пети.

Третье задание выполнено, если правильно указана позиция, выигрышная для Вани, и построено дерево всех партий, возможных при Ваниной стратегии. Должно быть явно сказано, что в этом дереве в каждой позиции, где должен ходить Петя, разобраны все возможные ходы, а для позиций, где должен ходить Ваня, – только ход, соответствующий стратегии, которую выбрал Ваня.

Во всех случаях стратегии могут быть описаны так, как это сделано в примере решения, или другим способом.

Указания по оцениванию	Баллы
Выполнены второе и третье задания. Первое задание выполнено полностью или частично. Здесь и далее в решениях допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному ответу.	3
Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий. Задание 3 выполнено полностью. Первое и второе задания выполнены полностью. Первое задание выполнено полностью или частично; для заданий 2 и 3 указаны правильные значения $S$ .	2
Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий. Первое задание выполнено полностью. Во втором задании правильно указано одно из двух возможных значений $S$ , и для этого значения указана и обоснована выигрышная стратегия Пети. Первое задание выполнено частично, и для одного из остальных заданий правильно указано значение $S$ . Для второго и третьего заданий правильно указаны значения $S$ .	1
Не выполнено ни одно из условий, позволяющих поставить 3, 2 или 1 балл	0
<i>Максимальный балл</i>	3

**C4** По каналу связи передаётся последовательность положительных целых чисел  $X_1, X_2, \dots$  все числа не превышают 1000, их количество заранее неизвестно. Каждое число передаётся в виде отдельной текстовой строки, содержащей десятичную запись числа. Признаком конца передаваемой последовательности является число 0.

Участок последовательности от элемента  $X_T$  до элемента  $X_{T+N}$  называется подъёмом, если на этом участке каждое следующее число больше или равно предыдущему, причем участок нельзя расширить, т.е.

1)  $T = 1$  или  $X_{T-1} > X_T$

2)  $X_{T+N}$  – последний элемент последовательности или  $X_{T+N} > X_{T+N+1}$ .

Высотой подъёма называется разность  $X_{T+N} - X_T$ . Подъём считается значительным, если высота подъёма больше величины минимального элемента этого подъёма.

Напишите эффективную программу, которая вычисляет количество значительных подъёмов в заданной последовательности.

Программа должна вывести результаты в следующей форме:

Получено чисел: ...

Найдено значительных подъёмов: ...

Размер памяти, которую использует программа, не должен зависеть от длины переданной последовательности чисел.

Перед текстом программы кратко опишите алгоритм решения задачи.

*Пример входных данных:*

```
144
17
21
27
3
7
9
11
25
0
```

*Пример выходных данных для приведённого выше примера входных данных:*

Получено чисел: 9

Найдено значительных подъёмов: 1

#### Содержание верного ответа и указания по оцениванию

(допускаются иные формулировки ответа, не искажающие его смысла)

Программа читает все входные данные один раз, не запоминая все входные данные в массиве. Во время чтения программа помнит число **up** – текущее количество значительных подъёмов, а также необходимые сведения о текущем участке неубывания, например, число **start** – значение первого (и, значит, минимального) элемента участка неубывания и последнее прочитанное число **last** (это число – наибольшее из чисел текущего участка неубывания). Прочитав очередное число **x**, программа сравнивает его с числом **last**. Если  $x < last$ , то фиксируется конец участка неубывания и начало нового участка. Если при этом выполнено условие  $last > 2 * start$ , нужно увеличить количество **up**. При обнаружении конца массива следует таким же способом проверить, является ли последний подъём значительным. Если нужно, следует увеличить значение **up**.

Ниже приведены примеры решения задания на языке Паскаль, на алгоритмическом языке и на языке Бейсик. Допускаются решения, записанные на других языках программирования.

**Пример правильной и эффективной программы на языке Паскаль:**

```
program c4_1;
var
  n, up, x, start, last : integer;
begin
  n:=0;
  up:=0;
  start:=1001;
  last:=1001;
  repeat
    readln(x);
    n:=n+1;
    if x < last then begin
      if last > 2*start then up:=up+1;
      start:=x;
    end;
    last:=x;
  until x = 0;
  writeln('Получено чисел: ', n);
  writeln('Найдено значительных подъёмов: ', up);
end.
```

**Пример правильной и эффективной программы на алгоритмическом языке:**

```
алг C4_1
нач
  цел n, up, x, start, last
  n:=0
  up:=0
  start:=1001
  last:=1001
  нц
    ввод x
    n:=n+1
    если x < last то
      если last > 2*start то up:=up+1 все
      start:=x;
    все
    last:=x;
  кц при x=0
  вывод "Получено чисел: ", n, нс
  вывод "Найдено значительных подъёмов: ", up, нс
кон
```

**Пример правильной и эффективной программы на языке Бейсик:**

```

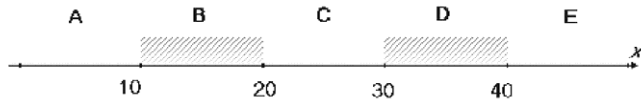
DIM n, up, x, start, last AS INTEGER
n=0
up=0
start=1001
last=1001
DO
  INPUT x
  n = n + 1
  IF x < last THEN
    IF last > 2 * start THEN up = up + 1
    start = x;
  END IF
  last = x
LOOP UNTIL x = 0
PRINT "Получено чисел: "; n
PRINT "Найдено значительных подъемов: "; up

```

Указания по оцениванию	Баллы
Программа правильно работает для любых входных данных произвольного размера и находит ответ, не сохраняя входные данные в массиве. Допускается наличие в тексте программы одной синтаксической ошибки: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных (если одна и та же ошибка встречается несколько раз, то это считается за одну ошибку).	4
Программа работает верно, но размер используемой памяти зависит от длины используемой последовательности. Например, входные данные запоминаются в массиве или другой структуре данных (например, контейнер <code>priority_queue</code> , <code>set</code> или <code>map</code> в C++), размер которого соответствует количеству прочитанных чисел. Допускается наличие от одной до трёх синтаксических ошибок. Возможно, в принципиально верно организованном вводе данных есть ошибка.	3
Программа работает в целом верно, эффективно или нет, но в реализации алгоритма содержатся ошибки при инициализации цикла анализа массива данных или обработке конца массива. Допускается наличие от одной до пяти синтаксических ошибок, описанных выше.	2
В программе есть блок выделения очередного участка возрастания, однако этот блок написан с ошибками. Допускается наличие от одной до пяти синтаксических ошибок, описанных выше	1
Прочее	0
<i>Максимальный балл</i>	4

**Критерии оценивания заданий с развёрнутым ответом**

**C1** Требовалось написать программу, при выполнении которой с клавиатуры считывается координата точки на прямой ( $x$  – действительное число) и определяется принадлежность этой точки заданной области (включая границы). Программист торопился и написал программу неправильно.

**Бейсик**

```
INPUT x
IF x<=30 THEN
  IF x<=20 THEN
    IF x<=10 THEN
      PRINT "не принадлежит"
    ELSE
      PRINT "принадлежит"
    END IF
  END IF
END IF
END IF
END
```

**Паскаль**

```
var x: real;
begin
  readln(x);
  if x<=30 then
    if x<=20 then
      if x<=10 then
        write('не принадлежит')
      else
        write('принадлежит')
      end if
    end if
  end if
end.
```

**Си**

```
#include <stdio.h>
void main()
{
  float x;
  scanf("%f",&x);
  if (x<=30)
    if (x<=20)
      if (x<=10)
        printf("не принадлежит");
      else
        printf("принадлежит");
  }
}
```

**Алгоритмический язык**

```
алг
нач
  вещ x
  ввод x
  если x<=30 то
    если x<=20 то
      если x<=10 то
        вывод 'не принадлежит'
      иначе
        вывод 'принадлежит'
      все
    все
  все
кон
```

Последовательно выполните следующее.

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (A, B, C, D и E). Точки, лежащие на границах областей (то есть, точки 10, 20, 30 и 40) при заполнении таблицы не рассматриваются.

Область	Условие 1 ( $x \leq 30$ )	Условие 2 ( $x \leq 20$ )	Условие 3 ( $x \leq 10$ )	Программа выведет	Область обрабатывается верно
A					
B					
C					
D					
E					

В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.», если программа ведет себя по-разному для разных значений, принадлежащих данной области. В столбце «Программа выведет» укажите, что программа выведет на экран. Если программа ничего не выводит, поставьте «—» (прочерк). Если для разных значений, принадлежащих области, будут выведены разные тексты, напишите «не изв.». 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.)

**Содержание верного ответа и указания по оцениванию**

(допускаются иные формулировки ответа, не искажающие его смысла)

Элементы ответа:

1.

Область	Условие 1 ( $x \leq 30$ )	Условие 2 ( $x \leq 20$ )	Условие 3 ( $x \leq 10$ )	Программа выведет	Область обрабатывается верно
A	да	да	да	не принадлежит	Да
B	да	да	нет	принадлежит	Да
C	да	нет	—	—	Нет
D	нет	—	—	—	Нет
E	нет	—	—	—	Нет

2. Возможная доработка (Паскаль):

```
if (x>=10) and (x<=20) or (x>=30) and (x<=40) then
  write('принадлежит')
else
  write('не принадлежит')
```

Возможны и другие способы доработки.

Например:

```
if x>=10 then
  if x<=20 then
    write('принадлежит')
  else
    if x>=30 then
      if x<=40 then
        write('принадлежит')
      else
        write('не принадлежит')
    else
      write('не принадлежит')
else
  write('не принадлежит')
```

Другой пример:

```
if abs(abs(x-25)-10)<=5 then
  write('принадлежит')
else
  write('не принадлежит')
```

**Обратите внимание!** В задаче требовалось выполнить три действия: указать для каждой области, как будет работать программа, что она выведет на экран и правильно ли это (в виде таблицы) и исправить две ошибки.

Баллы за данное задание начисляются как сумма баллов за верное выполнение каждого действия.

1. Верное заполнение предложенной таблицы.

2. Неправильное использование условного оператора, в результате чего при невыполнении первого или второго условия программа не выдавала ничего

(отсутствуют случаи ELSE). Исправлением этой ошибки может быть либо добавление случая ELSE к каждому условию IF, либо объединение всех условий IF в одно при помощи конъюнкции.

В сложных случаях это действие считается выполненным, если программа выдаёт одно из двух сообщений «принадлежит» или «не принадлежит» для любых чисел  $x$ , при этом программа не стала работать хуже, чем раньше, то есть для всех точек, для которых программа ранее выдавала верный ответ, доработанная программа также должна выдавать верный ответ.

3. Приведённых трёх ограничений недостаточно для описания двух областей (потеряно условие  $x \leq 40$ ). Кроме того, необходимо учесть, что области не соединены. Исправлением этой ошибки может быть разбиение области на две части и использование дизъюнкции, либо использование сложной (для выведения) математической конструкции ( $||x-25|-10| \leq 5$ ), либо использование сложной комбинации каскадных условий.

В сложных случаях это действие считается выполненным, если верно определены заштрихованные области, то есть программа выводит сообщение «принадлежит» для всех точек закрашенных областей, и только для них, для точек вне заштрихованных областей программа выводит «не принадлежит» или не выводит ничего.

Указания по оцениванию	Баллы
Правильно выполнены оба пункта задания. Исправлены две ошибки. Программа для всех чисел $x$ верно определяет принадлежность точки заштрихованной области. В работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения	3
1. Правильно выполнены два действия из трёх (исправлены обе ошибки, но в первом пункте задания не приведена таблица (либо таблица содержит ошибки более чем в одной строке), либо приведена таблица (которая содержит ошибки не более чем в одной строке), но исправлена только одна ошибка программы). При написании операций сравнения допускается одно неправильное использование строгих/нестрогих неравенств (считается несущественной ошибкой, погрешностью записи). Например, вместо « $x \leq 40$ » используется « $x < 40$ ». 2. Или выполнены все три действия, но при этом в логическом выражении неверно учтены приоритеты логических операций (не расставлены или неправильно расставлены скобки в выражениях)	2
Правильно выполнено только одно действие из трёх, то есть либо только приведена таблица, которая содержит ошибки в не более чем двух строках, либо таблица не приведена (или приведена и содержит ошибки более чем в двух строках), но исправлена одна ошибка программы. При оценивании этого задания на 1 балл допускается не учитывать корректность работы программ на точках границ областей (вместо нестрогих неравенств в решении были использованы строгие неравенства или наоборот)	1
Все пункты задания выполнены неверно (таблица анализа правильности алгоритма не приведена либо содержит три и более строк с ошибками, программа не приведена либо ни одна из двух ошибок не исправлена)	0
<i>Максимальный балл</i>	
	3

**C2** Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 1000. Элемент массива называется хорошим, если это двузначное число, причём цифра в разряде десятков больше, чем цифра в разряде единиц. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести сумму всех хороших элементов массива.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

#### Паскаль

```
const
  N=30;
var
  a: array [1..N] of integer;
  i, j, s: integer;
begin
  for i:=1 to N do
    readln(a[i]);
  ...
end.
```

#### Бейсик

```
N=30
DIM A(N) AS INTEGER
DIM I, J, S AS INTEGER
FOR I = 1 TO N
  INPUT A(I)
NEXT I
...
END
```

#### Си

```
#include <stdio.h>
#define N 30
void main(){
  int a[N];
  int i, j, s;
  for (i=0; i<N; i++)
    scanf("%d", &a[i]);
  ...
}
```

#### Алгоритмический язык

```
алг
нач
  цел N=30
  целтаб a[1:N]
  цел i, j, s
  нц для i от 1 до N
    ввод a[i]
  кц
  ...
кон
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, Free Pascal 2.4) или в виде блок-схемы. В этом случае вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

---

**Содержание верного ответа и указания по оцениванию**  
(допускаются иные формулировки ответа, не искажающие его смысла)

#### Пример программы на языке Паскаль

```
s:=0;
for i:=1 to N do begin
  if (10<=a[i]) and (a[i]<=99) and
    ((a[i] div 10) > (a[i] mod 10) ) then s:=s+a[i];
end;
writeln(s);
```

#### Пример программы на языке Бейсик

```
S = 0
FOR I = 1 TO N
  IF 10 <=A(I) AND A(I)<=99 AND A(I)\10 > A(I) MOD 10 THEN
    S = S + A(I)
  END IF
NEXT I
PRINT S
```

#### Пример программы на языке Си

```
s=0;
for (i=0; i<N; i++) {
  if (10<=a[i] && a[i]<=99 && a[i]/10 > a[i]%10)
    s = s + a[i];
}
printf("%d", s);
```

**Пример программы на алгоритмическом языке**

```

s := 0
нц для i от 1 до N
  если 10 <= a[i] <= 99 и div(a[i], 10) > mod(a[i], 10)
    то s := s + a[i]
  все
кц
вывод s

```

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Допускается запись алгоритма на другом языке, использующая аналогичные переменные. В случае, если язык программирования использует типизированные переменные, описания переменных должны быть аналогичны описаниям переменных в тексте задания. Использование нетипизированных или необъявленных переменных возможно только в случае, если это допускается языком программирования, при этом количество переменных и их идентификаторы должны соответствовать условию задачи. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы	2
В любом варианте решения может присутствовать не более одной ошибки из числа следующих: 1) Не инициализируется или неверно инициализируется переменная S. 2) Неверно осуществляется проверка того, что элемент массива – хороший. 3) Вместо проверки того, что элемент хороший, аналогичная проверка выполняется для индекса элемента. 4) Неверно осуществляется накопление суммы в цикле (например, s := a[i];). 5) Отсутствует вывод ответа. 6) Используется переменная, не объявленная в разделе описания переменных. 7) Не указано или неверно указано условие завершения цикла. 8) Индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно. 9) Неверно расставлены операторные скобки	1
Ошибок, перечисленных в п. 1–9, две или больше, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

**С3** Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **добавить в кучу один камень** или **возвести количество камней в квадрат**. Например, имея кучу из 7 камней, за один ход можно получить кучу из 8 или 49 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится 100 или более. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 100 или больше камней.

В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 99$ .

Говорят, 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) При каких значениях числа  $S$  Петя может выиграть первым ходом? Укажите все такие значения и выигрывающий ход Пети.

б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

2. Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причем (а) Петя не может выиграть первым ходом, но (б) Петя может выиграть своим вторым ходом, независимо от того, как будет ходить Ваня.

Для указанных значений  $S$  опишите выигрышную стратегию Пети.

3. Укажите такое значение  $S$ , при котором  
– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и при этом  
– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

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

**Содержание верного ответа и указания по оцениванию**

(допускаются иные формулировки ответа, не искажающие его смысла)

1. а) Петя может выиграть, если  $S = 10, \dots, 99$ . Пете достаточно возвести количество камней в квадрат. При  $S < 10$  получить за один ход 100 или больше камней невозможно.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет  $S = 9$  камней. Тогда после первого хода Пети в куче будет 10 камней или 81 камень. В обоих случаях Ваня возводит количество камней в квадрат и выигрывает в один ход.

2. Возможные значения  $S$ : 3, 8. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 9 камней (при  $S=3$  он возводит количество камней в квадрат; при  $S=8$  – добавляет 1 камень). Эта позиция разобрана в п. 1 б. В ней игрок, который будет ходить (теперь это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.

3. Возможное значение  $S$ : 7. После первого хода Пети в куче будет 8 или 49 камней. Если в куче станет 49 камней, Ваня возводит количество камней в квадрат и выигрывает своим первым ходом. Ситуация, когда в куче 8 камней разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Ваня) выигрывает своим вторым ходом.

В таблице изображено дерево возможных партий при описанной стратегии Вани. Заключительные позиции (в них выигрывает Ваня) подчеркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

Исх. положение	Положения после очередных ходов			
	1-й ход Пети (разобраны все ходы)	1-й ход Вани (только ход по стратегии)	2-й ход Пети (разобраны все ходы)	2-й ход Вани (только ход по стратегии)
7	$7+1=8$	$8+1=9$	$9+1=10$	$10*10=100$
	$7*7=49$	$49*49=2401$	$9*9=81$	$81*81=6561$

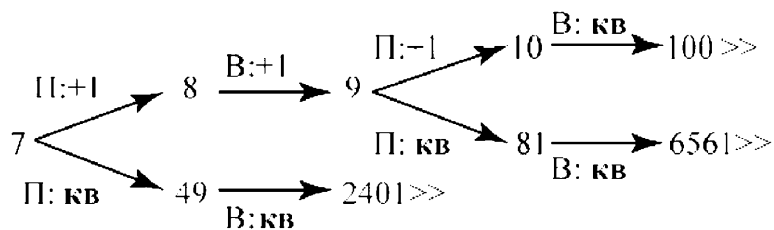


Рис. 1. Дерево всех партий, возможных при Ваниной стратегии. Знаком >> обозначены позиции, в которых партия заканчивается

В задаче от ученика требуется выполнить 3 задания. Их трудность возрастает. Количество баллов в целом соответствует количеству выполненных заданий (подробнее см. ниже).

Ошибка в решении, не искажающая основного замысла и не приведшая к неверному ответу, например, арифметическая ошибка при вычислении количества камней в заключительной позиции, при оценке решения не учитывается.

Первое задание считается выполненным полностью, если выполнены полностью оба пункта а) и б). Пункт а) считается выполненным полностью, если правильно указаны все позиции, в которых Петя выигрывает первым ходом, и указано, каким должен быть первый ход. Пункт б) считается выполненным, если правильно указана позиция, в которой Ваня выигрывает первым ходом, и описана стратегия Вани, т. е. показано, как Ваня может получить кучу, в которой содержится нужное количество камней, при любом ходе Пети.

Первое задание считается выполненным частично, если (а) правильно указаны все позиции, в которых Петя выигрывает первым ходом, (б) правильно указана позиция, в которой Ваня выигрывает первым ходом, и явно сказано, что при любом ходе Пети Ваня может получить кучу, которая содержит нужное для выигрыша количество камней. Отличие от полного решения в том, что явно не указаны ходы, ведущие к выигрышу.

Второе задание выполнено, если правильно указаны обе позиции, выигрышные для Пети, и описана соответствующая стратегия Пети – так, как это написано в примере решения, или другим способом, например, с помощью дерева всех партий, возможных при выбранной стратегии Пети.

Третье задание выполнено, если правильно указана позиция, выигрышная для Вани, и построено дерево всех партий, возможных при Ваниной стратегии. Должно быть явно сказано, что в этом дереве в каждой позиции, где должен ходить Петя, разобраны все возможные ходы, а для позиций, где должен ходить Ваня, – только ход, соответствующий стратегии, которую выбрал Ваня.

Во всех случаях стратегии могут быть описаны так, как это сделано в примере решения, или другим способом.

Указания по оцениванию	Баллы
Выполнены второе и третье задания. Первое задание выполнено полностью или частично. Здесь и далее в решениях допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному ответу.	3
Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий. Задание 3 выполнено полностью. Первое и второе задания выполнены полностью. Первое задание выполнено полностью или частично; для заданий 2 и 3 указаны правильные значения $S$ .	2
Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий. Первое задание выполнено полностью. Во втором задании правильно указано одно из двух возможных значений $S$ , и для этого значения указана и обоснована выигрышная стратегия Пети. Первое задание выполнено частично, и для одного из остальных заданий правильно указано значение $S$ . Для второго и третьего заданий правильно указаны значения $S$ .	1
Не выполнено ни одно из условий, позволяющих поставить 3, 2 или 1 балл	0
<i>Максимальный балл</i>	3



**C4** По каналу связи передаётся последовательность положительных целых чисел  $X_1, X_2, \dots$  все числа не превышают 1000, их количество заранее неизвестно. Каждое число передаётся в виде отдельной текстовой строки, содержащей десятичную запись числа. Признаком конца передаваемой последовательности является число 0.

Участок последовательности от элемента  $X_T$  до элемента  $X_{T+N}$  называется подъёмом, если на этом участке каждое следующее число больше или равно предыдущему, причем участок нельзя расширить, т.е.

1)  $T = 1$  или  $X_{T-1} > X_T$

2)  $X_{T+N}$  – последний элемент последовательности или  $X_{T+N} > X_{T+N+1}$ .

Высотой подъёма называется разность  $X_{T+N} - X_T$ . Подъём считается значительным, если высота подъёма больше величины минимального элемента этого подъёма.

Напишите эффективную программу, которая вычисляет количество значительных подъёмов в заданной последовательности.

Программа должна вывести результаты в следующей форме:

Получено чисел: ...

Найдено значительных подъёмов: ...

Размер памяти, которую использует программа, не должен зависеть от длины переданной последовательности чисел.

Перед текстом программы кратко опишите алгоритм решения задачи.

*Пример входных данных:*

144  
17  
21  
27  
3  
7  
9  
11  
25  
0

*Пример выходных данных для приведённого выше примера входных данных:*

Получено чисел: 9

Найдено значительных подъёмов: 1

#### Содержание верного ответа и указания по оцениванию

(допускаются иные формулировки ответа, не искажающие его смысла)

Программа читает все входные данные один раз, не запоминая все входные данные в массиве. Во время чтения программа помнит число **up** – текущее количество значительных подъёмов, а также необходимые сведения о текущем участке неубывания, например, число **start** – значение первого (и, значит, минимального) элемента участка неубывания и последнее прочитанное число **last** (это число – наибольшее из чисел текущего участка неубывания). Прочитав очередное число **x**, программа сравнивает его с числом **last**. Если  $x < last$ , то фиксируется конец участка неубывания и начало нового участка. Если при этом выполнено условие  $last > 2 * start$ , нужно увеличить количество **up**. При обнаружении конца массива следует таким же способом проверить, является ли последний подъём значительным. Если нужно, следует увеличить значение **up**.

Ниже приведены примеры решения задания на языке Паскаль, на алгоритмическом языке и на языке Бейсик. Допускаются решения, записанные на других языках программирования.

**Пример правильной и эффективной программы на языке Паскаль:**

```
program c4_1;
var
  n, up, x, start, last : integer;
begin
  n:=0;
  up:=0;
  start:=1001;
  last:=1001;
  repeat
    readln(x);
    n:=n+1;
    if x < last then begin
      if last > 2*start then up:=up+1;
      start:=x;
    end;
    last:=x;
  until x = 0;
  writeln('Получено чисел: ', n);
  writeln('Найдено значительных подъёмов: ', up);
end.
```

**Пример правильной и эффективной программы на алгоритмическом языке:**

```
алг C4_1
нач
  цел n, up, x, start, last
  n:=0
  up:=0
  start:=1001
  last:=1001
  нц
    ввод x
    n:=n+1
    если x < last то
      если last > 2*start то up:=up+1 все
      start:=x;
    все
    last:=x;
  кц при x=0
  вывод "Получено чисел: ", n, нс
  вывод "Найдено значительных подъёмов: ", up, нс
кон
```

**Пример правильной и эффективной программы на языке Бейсик:**

```

DIM n, up, x, start, last AS INTEGER
n=0
up=0
start=1001
last=1001
DO
  INPUT x
  n = n + 1
  IF x < last THEN
    IF last > 2 * start THEN up = up + 1
    start = x;
  END IF
  last = x
LOOP UNTIL x = 0
PRINT "Получено чисел: "; n
PRINT "Найдено значительных подъемов: "; up

```

Указания по оцениванию	Баллы
Программа правильно работает для любых входных данных произвольного размера и находит ответ, не сохраняя входные данные в массиве. Допускается наличие в тексте программы одной синтаксической ошибки: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных (если одна и та же ошибка встречается несколько раз, то это считается за одну ошибку).	4
Программа работает верно, но размер используемой памяти зависит от длины используемой последовательности. Например, входные данные запоминаются в массиве или другой структуре данных (например, контейнер <code>priority_queue</code> , <code>set</code> или <code>map</code> в C++), размер которого соответствует количеству прочитанных чисел. Допускается наличие от одной до трёх синтаксических ошибок. Возможно, в принципиально верно организованном вводе данных есть ошибка.	3
Программа работает в целом верно, эффективно или нет, но в реализации алгоритма содержатся ошибки при инициализации цикла анализа массива данных или обработке конца массива. Допускается наличие от одной до пяти синтаксических ошибок, описанных выше.	2
В программе есть блок выделения очередного участка возрастания, однако этот блок написан с ошибками. Допускается наличие от одной до пяти синтаксических ошибок, описанных выше	1
Прочее	0
<i>Максимальный балл</i>	4