

**Единый государственный экзамен  
по ИНФОРМАТИКЕ и ИКТ**

**Инструкция по выполнению работы**

Экзаменационная работа состоит из двух частей, включающих в себя 27 заданий. Часть 1 содержит 23 задания с кратким ответом, часть 2 содержит 4 задания с развёрнутым ответом.

На выполнение экзаменационной работы по информатике и ИКТ отводится 3 часа 55 минут (235 минут).

Ответы к заданиям 1–23 записываются в виде числа, последовательности букв или цифр. Ответы запишите в поле ответа в тексте работы, а затем перенесите в бланк ответов № 1.

КИМ

Ответ: 23

1	2	3																							
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Бланк

Задания 24–27 требуют развёрнутого ответа. В бланке ответов № 2 укажите номер задания и запишите его полное решение.

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

При выполнении заданий можно пользоваться черновиком. Записи в черновике не учитываются при оценивании работы.

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.

- Обозначения для логических связей (операций):
  - отрицание (инверсия, логическое НЕ) обозначается  $\neg$  (например,  $\neg A$ );
  - конъюнкция (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );
  - дизъюнкция (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );
  - следование (импликация) обозначается  $\rightarrow$  (например,  $A \rightarrow B$ );
  - 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);
  - символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).

2. Два логических выражения, содержащих 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  $A \wedge B$  неравносильны (значения выражений разные, например, при  $A = 1, B = 0$ ).

3. Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование), тождество. Таким образом,  $\neg A \wedge B \vee C \wedge D$  означает то же, что и  $((\neg A) \wedge B) \vee (C \wedge D)$ .

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. Обозначения Мбайт и Кбайт используются в традиционном для информатики смысле – как обозначения единиц измерения, чьё соотношение с единицей «байт» выражается степенью двойки.



**Часть 1**

*Ответами к заданиям 1–23 являются число, последовательность букв или цифр, которые следует записать в БЛАНК ОТВЕТОВ № 1 справа от номера соответствующего задания, начиная с первой клеточки, без пробелов, запятых и других дополнительных символов. Каждый символ пишете в отдельной клеточке в соответствии с приведёнными в бланке образцами.*

**1** Сколько единиц в двоичной записи восьмеричного числа  $7351_8$ ?  
 Ответ: \_\_\_\_\_.

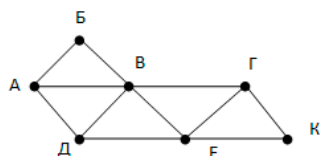
**2** Логическая функция F задаётся выражением  $(a \wedge \neg c) \vee (\neg a \wedge b \wedge c)$ .  
 Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных a, b, c.

???	???	???	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

В ответе напишите буквы a, b, c в том порядке, в котором идут соответствующие им столбцы (без разделителей).

Ответ: \_\_\_\_\_.

**3** На рисунке справа схема дорог Н-ского района изображена в виде графа, в таблице содержатся сведения о длинах этих дорог (в километрах).



	П1	П2	П3	П4	П5	П6	П7
П1			30		25		18
П2			17	12			
П3	30	17		23		34	15
П4		12	23			46	
П5	25						37
П6			34	46			18
П7	18		15		37	18	

Так как таблицу и схему рисовали независимо друг от друга, то нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите, какова длина дороги из пункта А в пункт Д.

Ответ: \_\_\_\_\_.

**4** В каталоге находятся файлы со следующими именами:  
 primera.dat  
 primera.doc  
 merchant.doc  
 k-mer.doc  
 omerta.doc  
 Tamerlan.docx

Определите, по какой из масок будет выбрано ровно два файла:  
 1) \*mer?\*.d\* 2) \*mer\*?.doc\*  
 3) ?\*mer?\*.doc 4) \*?mer\*?.doc\*

Ответ: \_\_\_\_\_.

**5** В сообщении встречается 10 разных букв. При его передаче использован неравномерный двоичный префиксный код. Известны коды трех букв: 11, 100, 101. Коды остальных семи букв имеют одинаковую длину. Какова минимальная суммарная длина всех 10 кодовых слов?

Ответ: \_\_\_\_\_.





**6** Автомат получает на вход трёхзначное число. По этому числу строится новое число по следующим правилам.

1. Перемножаются первая и вторая, а также вторая и третья цифры.
2. Полученные два числа записываются друг за другом в порядке неубывания без разделителей.

Пример. Исходное число: 631. Произведение:  $6 \cdot 3 = 18$ ;  $3 \cdot 1 = 3$ . Результат: 318. Укажите наибольшее число, при обработке которого автомат выдаёт результат 621.

Ответ: \_\_\_\_\_.

**7** Дан фрагмент электронной таблицы. Из одной из ячеек диапазона B1:B4 в одну из ячеек диапазона A1:A4 была скопирована формула. При этом адреса в формуле автоматически изменились и числовое значение в ячейке, куда производилось копирование, стало равным 230. В какую ячейку была скопирована формула? В ответе укажите только одно число – номер строки, в которой расположена ячейка.

	A	B	C	D	E
1		=D\$1+\$D1	1	10	100
2		=D\$2+\$D2	50	20	200
3		=D\$3+\$D3	150	30	300
4		=D\$4+\$D4	200	40	400

Ответ: \_\_\_\_\_.

**8** Запишите число, которое будет напечатано в результате выполнения следующей программы.

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; int main() { int n = 0, s = 0;   while (s &lt;= 365) {     s = s + 36;     n = n + 10;   }   printf("%d", n);   return 0;</pre>	<pre>n = 0 s = 0 while s &lt;= 365:   s = s + 36   n = n + 10 print(n)</pre>	<pre>var n, s: integer; begin   n:= 0;   s:= 0;   while s &lt;= 365 do   begin     s:= s + 36;     n:= n + 10   end;</pre>

```
}
write(n)
end.
```

Ответ: \_\_\_\_\_.

**9** Музыкальный фрагмент был оцифрован и записан в виде файла без использования сжатия данных. Получившийся файл был передан в город А по каналу связи за 96 секунд. Затем тот же музыкальный фрагмент был оцифрован повторно с разрешением в 4 раза выше и частотой дискретизации в 3 раза ниже, чем в первый раз. Сжатие данных не производилось. Полученный файл был передан в город Б за 16 секунд. Во сколько раз скорость пропускная способность канала в город Б больше пропускной способности канала в город А?

Ответ: \_\_\_\_\_.

**10** Все 5-буквенные слова, составленные из букв А, К, Р, У, записаны в алфавитном порядке. Вот начало списка:

1. ААААА
2. ААААК
3. ААААР
4. ААААУ
5. АААКА

...  
Какое количество слов находятся между словами РУКАА и УКАРА (включая эти слова)?

Ответ: \_\_\_\_\_.



**11** Ниже записаны две рекурсивные функции (процедуры): F и G. Сколько символов «звёздочка» будет напечатано на экране при выполнении вызова F(13)?

СИ	Python	Паскаль
<pre>void F(int n) {     if (n &gt; 0) G(n - 1); } void G(int n) {     printf("*");     if (n &gt; 1) {         printf("*");         F(n - 2);     } }</pre>	<pre>def F(n):     if n &gt; 0:         G(n - 1) def G(n):     print("*")     if n &gt; 1:         print("*")         F(n - 2)</pre>	<pre>procedure F(n: integer); begin     if n &gt; 0 then         G(n - 1); end; procedure G(n: integer); begin     writeln('*');     if n &gt; 1 then begin         writeln('*');         F(n - 2);     end; end;</pre>

Ответ: \_\_\_\_\_.

**12** Два узла, находящиеся в одной сети, имеют IP-адреса 61.58.73.42 и 61.58.75.136. Укажите наибольшее возможное значение третьего слева байта маски сети. Ответ запишите в виде десятичного числа.

Ответ: \_\_\_\_\_.

**13** При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 7 символов и содержащий только символы из 10-символьного набора: A, B, C, D, E, F, G, H, K, L. В базе данных для хранения сведений о каждом пользователе отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Кроме собственно пароля, для каждого пользователя в системе хранятся дополнительные сведения, для чего отведено 16 байт на одного пользователя. Определите объём памяти (в байтах), необходимый для хранения сведений о 20 пользователях.

Ответ: \_\_\_\_\_.

**14** Исполнитель Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах v и w обозначают цепочки цифр.

1. заменить (v, w)
2. нашлось (v)

Первая команда заменяет в строке первое слева вхождение цепочки v на цепочку w, вторая проверяет, встречается ли цепочка v в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь».

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 146 идущих подряд цифр 5? В ответе запишите полученную строку.

НАЧАЛО

ПОКА нашлось (333) ИЛИ нашлось (555)

ЕСЛИ нашлось (555)

ТО заменить (555, 3)

ИНАЧЕ заменить (333, 5)

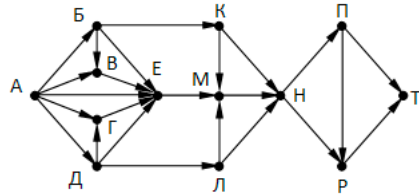
КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

Ответ: \_\_\_\_\_.

- 15 На рисунке – схема дорог, связывающих города А, Б, В, Г, Д, Е, К, Л, М, Н, П, Р, С, Х, Т. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей, ведущих из города А в город Т?



Ответ: \_\_\_\_\_.

- 16 Сколько значащих нулей в двоичной записи числа

$$4^{512} + 8^{512} - 2^{128} - 250$$

Ответ: \_\_\_\_\_.

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

Ключевое слово	Количество страниц(тыс.)
Математика&Информатика	330
Математика&Физика	270
Математика&(Информатика/Физика)	520

Какое количество страниц (в тысячах) будет найдено по запросу *Математика & Информатика & Физика*?

Ответ: \_\_\_\_\_.

- 18 Обозначим делится без остатка на натуральное число  $m$ ». Для какого наименьшего натурального числа А формула

$$\text{ДЕЛ}(x, A) \rightarrow (\text{ДЕЛ}(x, 21) \vee \text{ДЕЛ}(x, 35))$$

тождественно истинна (то есть принимает значение 1 при любом натуральном значении переменной  $x$ )?

Ответ: \_\_\_\_\_.

- 19 Ниже представлен фрагмент программы, обрабатывающей одномерный целочисленный массив с индексами от 0 до 10. В начале выполнения этого фрагмента в массиве находились двухзначные натуральные числа. Какое наибольшее значение может иметь переменная  $s$  после выполнения данной программы?

СИ	Python	Паскаль
<pre>s = 0; n = 10; for (i=0; i&lt;n-1; i++)     s=s+A[i]-A[i+2];</pre>	<pre>s = 0 n = 10 for i in range(n-1):     s=s+A[i]-A[i+2]</pre>	<pre>s := 0; n := 10; for i:=0 to n-2 do begin     s:=s+A[i]-A[i+2] end;</pre>

Ответ: \_\_\_\_\_.





**20** Ниже записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает число  $M$ . Известно, что  $x > 100$ . Укажите наименьшее такое (т.е. большее 100) число  $x$ , при вводе которого алгоритм печатает 35.

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; void main() {     int x, L, M;     scanf("%d", &amp;x);     L = x - 15;     M = x + 20;     while (L != M) {         if(L &gt; M)             L = L - M;         else             M = M - L;     }     printf("%d", M); }</pre>	<pre>x = int(input()) L = x - 15 M = x + 20 while L != M:     if L &gt; M:         L = L - M     else:         M = M - L print(M)</pre>	<pre>var x, L, M: integer; begin     readln(x);     L := x - 21;     M := x + 12;     while L &lt;&gt; M do         if L &gt; M then             L := L - M         else             M := M - L;     writeln(M); end.</pre>

Ответ: \_\_\_\_\_.

**21** Определите, количество чисел  $k$ , для которых следующая программа выведет такой же результат, что и для  $k = 10$ ?

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; long f(long n) {     return n * n * n; } int main() {     long k, i;     scanf("%ld", &amp;k);     i = 1;     while ( f(i) &lt; k )         i++;     if (f(i)-k &lt;= k-f(i-1))         printf("%ld", i);     else         printf("%ld", i-1);     return 0; }</pre>	<pre>def f(n):     return n * n * n k = int(input()) i = 1 while f(i) &lt; k:     i += 1 if f(i)-k &lt;= k-f(i-1):     print(i) else:     print(i-1)</pre>	<pre>var k, i : longint; function f(n: longint):     longint; begin     f := n * n * n; end; begin     readln(k);     i := 1;     while f(i) &lt; k do         i := i + 1;     if f(i)-k &lt;= k-f(i-1) then         writeln(i)     else writeln(i-1); end.</pre>

Ответ: \_\_\_\_\_.

**22** Исполнитель Калькулятор преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Умножить на 3

Программа для исполнителя Калькулятор – это последовательность команд. Сколько существует программ, для которых при исходном числе 4 результатом является число 50 и при этом траектория вычислений содержит число 6 и не содержит числа 12?

Ответ: \_\_\_\_\_.

**23** Сколько существует различных наборов значений логических переменных  $x_1, x_2, \dots, x_5, y_1, y_2, \dots, y_5, z_1, z_2, \dots, z_5$ , которые удовлетворяют всем перечисленным ниже условиям?

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

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

$$(z_1 \rightarrow z_2) \wedge (z_2 \rightarrow z_3) \wedge (z_3 \rightarrow z_4) \wedge (z_4 \rightarrow z_5) = 1$$

$$x_5 \wedge y_5 \wedge z_5 = 0$$

Ответ: \_\_\_\_\_.

**Не забудьте перенести все ответы в бланк ответов № 1 в соответствии с инструкцией по выполнению работы.**

Часть 2

Для записи ответов на задания этой части (24–27) используйте **БЛАНК ОТВЕТОВ № 2**. Запишите сначала номер задания (24, 25 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

**24** Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число  $N$ , не превосходящее  $10^9$ , и выводится сумма цифр этого числа. Программист написал программу неправильно.

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; int main() {     int N, sum, d;     scanf("%d", &amp;N);     sum = 1;     while ( N &gt; 0 ) {         d = N % 10;         N = N / 10;         sum = d;     }     printf("%d", sum);     return 0; }</pre>	<pre>N = int(input()) sum = 1 while N &gt; 0:     d = N % 10     N = N // 10     sum = d print(sum)</pre>	<pre>var N: longint;     sum, d: integer; begin     readln(N);     sum := 1;     while N &gt; 0 do begin         d := N mod 10;         N := N div 10;         sum := d;     end;     writeln(sum); end.</pre>

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 256.
2. Приведите пример такого трёхзначного числа  $X$ , при вводе которого программа выведет правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:
  - 1) выпишите строку, в которой сделана ошибка;
  - 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

**25** Дан массив, содержащий 40 положительных целых чисел. Напишите на одном из языков программирования программу, которая находит в этом массиве количество локальных минимумов. Локальным минимумом называется элемент массива, который меньше всех своих соседей. Например, в массиве из 6 элементов, содержащем числа 4, 6, 12, 7, 3, 8, есть два локальных минимума: это элементы, равные 4 и 3. Программа должна вывести общее количество подходящих элементов, значения элементов выводить не нужно.

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; #define n 40 int main() {     int a[n];     int i, j, k;     for (i = 0; i &lt; n; i++)         scanf("%d", &amp;a[i]);     ...     return 0; }</pre>	<pre># допускается также # использовать две # целочисленные # переменные j, k a = [] n = 40 for i in range(n):     a.append(int(input())) ...</pre>	<pre>const n = 40; var     a: array [0..n-1]         of integer;     i, j, k: integer; begin     for i := 0 to n-1 do         readln(a[i]);     ... end.</pre>

**26** Два игрока, Паша и Вася, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Паша. За один ход игрок может добавить в кучу три камня или увеличить количество камней в куче в два раза и затем добавить в кучу один камень. Игра завершается в тот момент, когда количество камней в куче становится не менее 85. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 85 или больше камней. В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 84$ .

Задание 1. а) Укажите все такие значения числа  $S$ , при которых Паша может выиграть в один ход. Обоснуйте, что найдены все нужные значения  $S$ , и укажите выигрышающий ход для каждого указанного значения  $S$ . б) Укажите такое значение  $S$ , при котором Паша не может выиграть за один ход, но при любом ходе Паши Вася может выиграть своим первым ходом. Опишите выигрышную стратегию Васи.

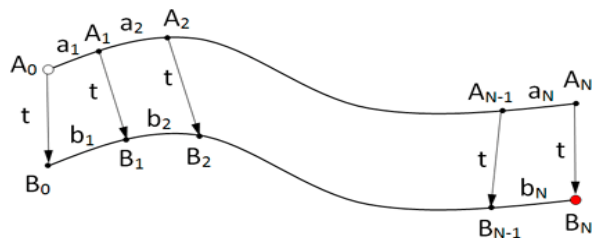
Задание 2. Укажите значения  $S$ , при которых у Паши есть выигрышная стратегия, причём Паша не может выиграть за один ход и может выиграть своим вторым ходом независимо от того, как будет ходить Вася. Для каждого указанного значения  $S$  опишите выигрышную стратегию Паши.





Задание 3. Укажите хотя бы одно значение  $S$ , при котором у Васи есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Паши, и у Васи нет стратегии, которая позволит ему гарантированно выиграть первым ходом. Для указанного значения  $S$  опишите выигрышную стратегию Васи. Постройте дерево всех партий, возможных при этой выигрышной стратегии Васи (в виде рисунка или таблицы).

27 Гоночная трасса состоит из двух основных дорог и нескольких переездов, позволяющих перейти с одной дороги на другую.



На всех участках, включая переезды, движение разрешено только в одну сторону, поэтому переезд возможен только с дороги А на дорогу В. Гонщик стартует в точке  $A_0$  и должен финишировать в точке  $B_N$ . Он знает, за какое время сможет пройти каждый участок пути по каждой дороге, то есть время прохождения участков  $A_0A_1, A_1A_2, \dots, A_{N-1}A_N, B_0B_1, B_1B_2, \dots, B_{N-1}B_N$ . Время прохождения всех переездов  $A_0B_0, A_1B_1, \dots, A_NB_N$  одинаково и известно гонщику. Необходимо определить, за какое минимальное время гонщик сможет пройти трассу.

В первой строке задаётся количество участков трассы  $N$ . Во второй строке задаётся целое число  $t$  – время (в секундах) прохождения каждого из переездов  $A_0B_0, A_1B_1, \dots, A_NB_N$ . В каждой из последующих  $N$  строк записано два целых числа  $a_i$  и  $b_i$ , задающих время (в секундах) прохождения очередного участка на каждой из дорог. В первой из этих строк указывается время прохождения участков  $A_0A_1$  и  $B_0B_1$ , во второй –  $A_1A_2$  и  $B_1B_2$  и т. д.

Пример входных данных:

3  
20

320 150  
200 440  
300 210

Пример выходных данных для приведённого выше примера входных данных:

750

<b>СОСТАВИТЕЛЬ ВАРИАНТА:</b>	
<b>ФИО:</b>	Константин Поляков
<b>Предмет:</b>	Информатика
<b>Аккаунт ВК:</b>	<a href="https://vk.com/kpolyakov_spb">https://vk.com/kpolyakov_spb</a>
<b>Сайт и доп. информация:</b>	<a href="http://kpolyakov.spb.ru/">http://kpolyakov.spb.ru/</a>

**О проекте «Пробный ЕГЭ каждую неделю»**

Данный ким составлен командой всероссийского волонтерского проекта «ЕГЭ 100 баллов» <https://vk.com/ege100ballov> и безвозмездно распространяется для любых некоммерческих образовательных целей.

**Нашли ошибку в варианте?**

**Напишите нам, пожалуйста, и мы обязательно её исправим!**

Для замечаний и пожеланий: [https://vk.com/topic-10175642\\_39008096](https://vk.com/topic-10175642_39008096)  
(также доступны другие варианты для скачивания)





Система оценивания экзаменационной работы по информатике и ИКТ

Часть 1

№ задания	Ответ
1	8
2	acb
3	46
4	3
5	36
6	732
7	3
8	110
9	8
10	121
11	9
12	252
13	400
14	553355
15	33
16	519
17	80
18	21
19	178
20	120
21	13
22	6
23	91

Часть 2

24 Требовалось написать программу, при выполнении которой с клавиатуры считывается натуральное число  $N$ , не превосходящее  $10^9$ , и выводится сумма цифр этого числа. Программист написал программу неправильно.

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; int main() {     int N, sum, d;     scanf("%d", &amp;N);     sum = 1;     while ( N &gt; 0 ) {</pre>	<pre>N = int(input()) sum = 1 while N &gt; 0:     d = N % 10     N = N // 10     sum = d print(sum)</pre>	<pre>var N: longint;     sum, d: integer; begin     readln(N);     sum := 1;     while N &gt; 0 do begin         d := N mod 10;</pre>

<pre>d = N % 10; N = N / 10; sum = d; } printf("%d", sum); return 0; }</pre>		<pre>N := N div 10; sum := d; end; writeln(sum); end.</pre>
--	--	---

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 256.
2. Приведите пример такого трёхзначного числа  $X$ , при вводе которого программа выведет правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:
  - 1) выпишите строку, в которой сделана ошибка;
  - 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	
<ol style="list-style-type: none"> <li>1. При вводе числа 256 программа выведет число 2.</li> <li>2. Программа выводит правильный ответ, если сумма цифр равна старшей цифре числа, например, для числа 300.</li> <li>3. В программе есть две ошибки.                             <ol style="list-style-type: none"> <li>1) Неверная инициализация. Строка с ошибкой:  <code>sum := 1;</code>                              Верное исправление:  <code>sum := 0;</code></li> <li>2) Неверное изменение переменной sum. Строка с ошибкой:  <code>sum := d;</code>                              Верное исправление:  <code>sum := sum + d;</code></li> </ol> </li> </ol>	
Указания по оцениванию	
В задаче требуется выполнить <b>три</b> действия. 1. Указать результат программы при данном вводе. Это действие считается выполненным, если указан верный результат работы программы при заданных входных данных. Экзаменуемый не обязан объяснять, как получен этот	Баллы



<p>результат, достаточно указать верное число.                  2. Указать пример ввода, при котором программа выводит верный ответ.                  Это действие считается выполненным, если указан пример числа, при вводе которого выводится верное сообщение (верный показатель степени или текст «Не существует», если введённое число не является степенью). Ученик не обязан указывать, что будет выведено, и объяснять, как работает программа.                  3. Найти и исправить ошибки в программе.                  Это действие считается выполненным, если верно указаны обе ошибки и предложены верные варианты исправления, при этом никакие верные строки программы не указаны в качестве неверных. В исправленной строке допускаются незначительные синтаксические ошибки (лишние или пропущенные знаки препинания, неточные написания служебных слов языка). Ошибка считается исправленной, если выполнены оба следующих условия:                  а) правильно указана строка с ошибкой;                  б) указан такой новый вариант строки, что при исправлении другой ошибки получается правильная программа</p>	
<p>Выполнены все три необходимых действия, и ни одна верная</p>	3
<p>Не выполнены условия, позволяющие поставить 3 балла.                  Имеет место одна из следующих ситуаций.                  1. Выполнены два первых действия, найдена и исправлена одна ошибка в программе, ни одна верная строка не названа ошибочной.                  2. Выполнены два первых действия, найдены и исправлены две ошибки в программе, одна верная строка названа ошибочной.                  3. Выполнено одно из первых двух действий, найдены и исправлены две ошибки в программе, ни одна верная строка не названа ошибочной</p>	2
<p>Не выполнены условия, позволяющие поставить 2 или 3 балла.                  При этом имеет место один из следующих случаев.                  1. Выполнены два первых действия. При этом несущественно, насколько правильно выполнено третье действие.                  2. Найдены и исправлены две ошибки в программе, не более чем одна верная строка названа ошибочной. При этом несущественно, насколько правильно выполнены действия 1 и 2.</p>	1

3. Выполнено одно из двух первых действий. Исправлена одна из двух ошибок. Не более чем одна верная строка названа ошибочной	
Не выполнены условия, позволяющие поставить 1, 2 или 3 балла	0
<i>Максимальный балл</i>	
	3

**25** Дан массив, содержащий 40 положительных целых чисел. Напишите на одном из языков программирования программу, которая находит в этом массиве количество локальных минимумов. Локальным минимумом называется элемент массива, который меньше всех своих соседей. Например, в массиве из 6 элементов, содержащем числа 4, 6, 12, 7, 3, 8, есть два локальных минимума: это элементы, равные 4 и 3. Программа должна вывести общее количество подходящих элементов, значения элементов выводить не нужно.

СИ	Python	Паскаль
<pre>#include &lt;stdio.h&gt; #define n 40 int main() {     int a[n];     int i, j, k;     for (i = 0; i &lt; n; i++)         scanf("%d", &amp;a[i]);     ...     return 0; }</pre>	<pre># допускается также # использовать две # целочисленные # переменные j, k a = [] n = 40 for i in range(n):     a.append(int(input())) ...</pre>	<pre>const n = 40; var a: array [0..n-1] of integer; i, j, k: integer; begin for i := 0 to n-1 do     readln(a[i]); ... end.</pre>

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

Паскаль	Python	Си
<pre>k := 0; if a[0]&lt;a[1] then k:=k+1; for i:=1 to n-2 do if (a[i-1]&gt;a[i]) and</pre>	<pre>k = 0 if a[0]&lt;a[1]:     k = k + 1; for i in range(n- 1):</pre>	<pre>k = 0; if (a[0]&lt;a[1]) k++; for i in range(n-1): if (a[i-1]&gt;a[i] &amp;&amp; a[i]&lt;a[i+1])</pre>



<pre>(a[i]&lt;a[i+1]) then   k:=k+1; if a[n-2]&gt;a[n-1] then   k:=k+1; writeln(k);</pre>	<pre>if (a[i-1]&gt;a[i] and   a[i]&lt;a[i+1]):   k += 1 if a[n-2]&gt;a[n-1]:   k = k + 1 print(k)</pre>	<pre>k++; if (a[n-2]&gt;a[n-1]) k++; printf("%d", k);</pre>
---	---	---

Указания по оцениванию	Баллы
<p><i>Общие указания</i></p> <p>1. В алгоритме, записанном на языке программирования, допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора программы.</p> <p>2. Эффективность алгоритма не имеет значения и не оценивается.</p> <p>3. Допускается запись алгоритма на языке программирования, отличном от языков, перечисленных в условии. В этом случае должны использоваться переменные, аналогичные описанным в условии. Если язык программирования использует типизированные переменные, описания переменных должны быть аналогичны описаниям переменных на естественном языке. 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</p>	
Предложен правильный алгоритм, выдающий в качестве результата верное значение	2
<p>Не выполнены условия, позволяющие поставить 2 балла. Предложено в целом верное решение, содержащее не более одной ошибки из числа следующих:</p> <ol style="list-style-type: none"> <li>1) в цикле происходит выход за границу массива (например, при использовании цикла от 1 до N);</li> <li>2) не инициализируется или неверно инициализируется счётчик количества найденных пар;</li> <li>3) счётчик количества пар в цикле не изменяется или изменяется неверно;</li> <li>4) неверно выделяется последняя цифра числа;</li> <li>5) при проверке выполнения условия для пары элементов используются неверные индексы;</li> </ol>	1

<ol style="list-style-type: none"> <li>6) последняя цифра выделяется не у самих элементов массива, а у их индексов;</li> <li>7) в сложном логическом условии простые проверки верны, но условие в целом построено неверно (например, перепутаны операции «И» и «ИЛИ», неверно расставлены скобки в логическом выражении);</li> <li>8) отсутствует вывод ответа;</li> <li>9) используется переменная, не объявленная в разделе описания переменных;</li> <li>10) не указано или неверно указано условие завершения цикла;</li> <li>11) индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно;</li> <li>12) неверно расставлены операторные скобки</li> </ol>	
Не выполнены условия, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл</i>	2

- 26** Два игрока, Паша и Вася, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Паша. За один ход игрок может добавить в кучу три камня или увеличить количество камней в куче в два раза и затем добавить в кучу один камень. Игра завершается в тот момент, когда количество камней в куче становится не менее 85. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 85 или больше камней. В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 84$ .
- Задание 1. а) Укажите все такие значения числа  $S$ , при которых Паша может выиграть в один ход. Обоснуйте, что найдены все нужные значения  $S$ , и укажите выигрывающий ход для каждого указанного значения  $S$ . б) Укажите такое значение  $S$ , при котором Паша не может выиграть за один ход, но при любом ходе Паши Вася может выиграть своим первым ходом. Опишите выигрившую стратегию Васи.
- Задание 2. Укажите все значения  $S$ , при которых у Паши есть выигрившая стратегия, причём Паша не может выиграть за один ход и может выиграть своим вторым ходом независимо от того, как будет ходить Вася. Для каждого указанного значения  $S$  опишите выигрившую стратегию Паши.
- Задание 3. Укажите хотя бы одно значение  $S$ , при котором у Васи есть выигрившая стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Паши, и у Васи нет стратегии, которая позволит ему гарантированно выиграть первым ходом. Для указанного значения  $S$



опишите выигрышную стратегию Васи. Постройте дерево всех партий, возможных при этой выигрышной стратегии Васи (в виде рисунка или таблицы).

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	
<p><b>Задание 1.</b> а) <math>S = 42 \dots 84</math> б) <math>S = 39, 40, 41</math>.</p> <p><b>Задание 2.</b> <math>S = 19, 20, 36, 37, 38</math>.</p> <p><b>Задание 3.</b> <math>S = 33, 34, 35</math>.</p>	
Указания по оцениванию	Баллы
<p><i>Предварительные замечания</i>                      В задаче от ученика требуется выполнить три 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.                      Пункт 1а считается выполненным, если правильно указаны все позиции, в которых Паша выигрывает первым ходом, и указано, каким должен быть первый ход. Пункт 1б считается выполненным, если (i) правильно указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) описаны выигрышные стратегии – так, как это сделано в образце решения, или другим способом. Первое задание считается выполненным полностью, если выполнены полностью оба пункта: 1а и 1б.                      Замечание для проверяющего. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника (см. условие задачи). Есть два основных способа сделать это. (1) Можно построить дерево всех партий, возможных при выбранной стратегии, и убедиться, что все заключительные позиции являются выигрышными для игрока, реализующего стратегию. (2) Можно свести задачу к рассмотренным выше позициям. Например, выигрышную стратегию для игрока, который ходит первым, можно описать, указав ход, ведущий в позицию, для которой известна выигрышная стратегия для игрока, который ходит вторым.</p>	

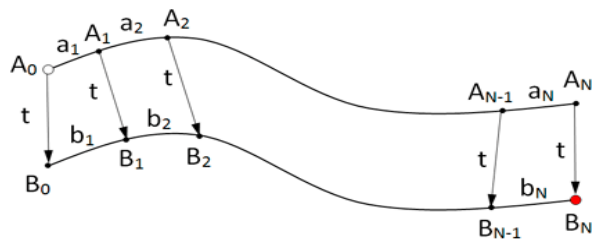
<p>Чтобы подобным образом описать выигрышную стратегию для игрока, который ходит вторым (Вали), нужно перебрать все возможные первые ходы Паши и убедиться, что для всех полученных позиций мы знаем выигрышную стратегию для игрока, который ходит первым.                      В примере решения мы используем в основном второй способ описания стратегии. Экзаменуемый может описывать стратегию любым удобным ему способом. Существенно (повторим), чтобы (1) для каждой позиции, которая может встретиться игроку, реализующему стратегию, было понятно, какой ход он должен сделать, и (2) было показано, что все возможные заключительные позиции выигрышные для этого игрока.</p> <p><i>Задание 2 считается выполненным, если (i) правильно указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) описаны выигрышные стратегии.</i></p> <p><i>Задание 3 считается выполненным, если (i) правильно указано, что выигрышную стратегию имеет Валя; (ii) правильно описано дерево всех партий, возможных при этой выигрышной стратегии (в виде рисунка или таблицы). При этом допускаются арифметические ошибки, не искажающие сути решения.</i>  <i>Во всех случаях стратегии могут быть описаны так, как это сделано в примере решения, или другим способом</i></p>	
<p>Выполнены второе и третье задания.                      Для первого задания правильно перечислены позиции, в которых Паша выигрывает первым ходом (п. 1а), и правильно указано, кто из игроков имеет выигрышную стратегию при указанных значениях <math>S</math> (п. 1б). При этом допускаются недочёты следующих типов:</p> <ul style="list-style-type: none"> <li>- в п. 1а не указано, каким ходом выигрывает Паша;</li> <li>- в п. 1б не указано, что игрокам нет смысла удваивать количество камней в куче.</li> </ul> <p>Здесь и далее в решениях допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному ответу</p>	3
<p>Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий.</p> <ul style="list-style-type: none"> <li>• Выполнено третье задание.</li> </ul>	2



<ul style="list-style-type: none"> <li>• Выполнены первое и второе задания.</li> <li>• Первое задание выполнено, возможно, при наличии недочётов, указанных в критериях на 3 балла; для второго задания (i) правильно указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) правильно указан первый ход Паши при выигрышной стратегии, однако не указано, что после выбранного хода Паши получается позиция, выигрышная для Вали; для третьего задания правильно указан игрок, имеющий выигрышную стратегию</li> </ul>	
<p>Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий.</p> <ul style="list-style-type: none"> <li>• Первое задание выполнено, возможно, с недочётами, указанными в критериях на 3 балла.</li> <li>• Второе задание выполнено, возможно, с недочётами, указанными в критериях на 2 балла.</li> <li>• Для второго и третьего заданий во всех случаях правильно указан игрок, имеющий выигрышную стратегию</li> </ul>	1
<p>Не выполнено ни одно из условий, позволяющих поставить 1, 2 или 3 балла</p>	0
<i>Максимальный балл</i>	3

27

Гонимая трасса состоит из двух основных дорог и нескольких переездов, позволяющих перейти с одной дороги на другую.



На всех участках, включая переезды, движение разрешено только в одну сторону, поэтому переезд возможен только с дороги А на дорогу В. Гонщик стартует в точке A0 и должен финишировать в точке BN. Он знает, за какое время сможет пройти каждый участок пути по каждой дороге, то есть время прохождения участков A0A1, A1A2, ..., AN-1AN, B0B1, B1B2,

..., BN-1BN. Время прохождения всех переездов A0B0, A1B1, ..., ANBN одинаково и известно гонщику. Необходимо определить, за какое минимальное время гонщик сможет пройти трассу.

В первой строке задаётся количество участков трассы N. Во второй строке задаётся целое число t – время (в секундах) прохождения каждого из переездов A0B0, A1B1, ..., ANBN. В каждой из последующих N строк записано два целых числа ai и bi, задающих время (в секундах) прохождения очередного участка на каждой из дорог. В первой из этих строк указывается время прохождения участков A0A1 и B0B1, во второй – A1A2 и B1B2 и т. д.

Пример входных данных:

```
3
20
320 150
200 440
300 210
```

Пример выходных данных для приведённого выше примера входных данных:

750

<p><b>Содержание верного ответа и указания по оцениванию</b> (допускаются иные формулировки ответа, не искажающие его смысла)</p>
<pre>var N, i, t, TA, TB, ai, bi: integer; begin   Readln( N );   Readln( t );   TA := 0;   TB := 0;   for i:=1 to N do begin     Readln( ai, bi );     TA := TA + ai;     if TB+bi &lt; TA+t then       TB:= TB + bi     else TB:= TA + t;</pre>





end; writeln(TB); end.	
Указания по оцениванию	Баллы
<p><b>Предварительные замечания.</b></p> <p>1. В задаче есть два задания (А и Б). Соответственно, ученик может представить две программы. В каждой из программ должно быть указано, решением какого из заданий она является. Если в работе представлена одна программа, то в ней также должно быть указано, решением какого из заданий она является.</p> <p>2. Если ученик не указал, к какому заданию относится программа, или можно предположить, что ученик ошибся в идентификации программ, необходимо следовать приведённым ниже инструкциям. Случай 2.1. Ученик представил только одну программу. Следует рассматривать программу как решение задания Б и оценивать её по соответствующим критериям.</p> <p>Случай 2.2. Ученик представил две программы, но указание задания есть только для одной из программ. Следует рассматривать вторую программу как ответ на оставшееся задание.</p> <p>Случай 2.3. Ученик представил две программы; ни для одной из них задание не указано, или в обоих решениях указано одно и то же задание. Следует первую (по порядку в представленных учеником материалах) программу рассматривать как ответ на задание А, а вторую – как ответ на задание Б.</p> <p>Случай 2.4. Ученик представил более двух программ. Следует рассматривать только две последние программы и соотносить их с заданиями по правилам 2.1–2.3.</p> <p>Случай 2.5. Решение, представленное в качестве решения задания А, по критериям для задания Б может быть оценено в 3 или 4 балла. При этом решение, представленное в качестве решения задания Б, оценено меньшим баллом. Следует считать, что ученик перепутал обозначения заданий и оценивать решение, представленное как решение задания А, по критериям задания Б.</p> <p><i>НАПОМИНАЕМ! Итоговый балл за задачу – это больший из баллов, полученных учеником за каждое из двух представленных решений.</i></p> <p><i>Пояснения для проверяющих.</i></p>	

<p>1. Задание Б является усложнением задания А. Если в качестве решения задания Б представлено решение задания А, то считается, что учеником допущена опечатка, и решение оценивается по критериям для задания А. В качестве решения задания А может быть представлена программа, которая решает задачу при произвольном количестве входных данных (как в задании Б), осуществляя полный перебор всех вариантов. Такая программа неэффективна по времени, поэтому она должна оцениваться 2 баллами.</p> <p>2. Два задания (и, соответственно, возможность для экзаменуемого представить две программы) дают ученику возможность (при его желании) сначала написать менее сложное и менее эффективное решение (задание А), которое даёт ему право получить 2 балла, а затем приступить к поиску более эффективного решения.</p> <p>3. Приведённые в п. 2.1–2.5 правила имеют целью избежать снижения баллов из-за того, что ученик перепутал обозначения заданий.</p> <p><i>Общие принципы оценивания решений</i></p> <p>4 балла ставится за эффективную и правильно работающую программу, решающую задачу в общем случае (задача Б). При этом программа может содержать до трёх синтаксических ошибок («описок»).</p> <p>3 балла ставится в случае, когда фактически задача решена и решена эффективно по времени, возможно, с хранением всех входных данных в массиве, но количество «описок» более трёх (но не более пяти) и допущено не более одной содержательной ошибки, не позволяющей усомниться в том, что экзаменуемый правильно придумал алгоритм.</p> <p>2 балла ставится, если программа, решающая задачу Б, в дополнение к неточностям, которые перечислены выше, работает неэффективно по времени и/или допущено до трёх упомянутых выше содержательных ошибок. Количество допустимых «описок» – до семи.</p> <p>2 балла также ставится за правильное решение упрощённой задачи (задача А).</p> <p>1 балл ставится, если программа написана неверно, но из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи.</p>	
---	--



Далее эти общие принципы уточнены	
<b>Критерии оценивания задания А</b>	
При решении задачи А программа верно находит требуемую сумму для любых 6 пар исходных данных. Допускается до пяти синтаксических и приравненных к ним ошибок (см. критерии оценивания задания Б на 4 балла)	2
Не выполнены условия, позволяющие поставить 2 балла. Из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи. Допускается любое количество «описок»	1
Не выполнены критерии, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл для задания А</i>	2
<b>Критерии оценивания задания Б</b>	
Программа правильно работает для любых соответствующих условию входных данных и при этом эффективна как по времени, так и по памяти, т.е. не используются массивы и другие структуры данных (в том числе стек рекурсивных вызовов), размер которых зависит от количества входных элементов, а время работы пропорционально этому количеству. Возможно использование массивов и динамических структур данных при условии, что в них в каждый момент времени хранится фиксированное количество элементов, требующих для хранения меньше 1Кб. Программа может содержать не более трёх синтаксических ошибок следующих видов: 1) пропущен или неверно указан знак пунктуации; 2) неверно написано или пропущено зарезервированное слово языка программирования; 3) не описана или неверно описана переменная; 4) применяется операция, недопустимая для соответствующего типа данных. К синтаксическим ошибкам приравнивается использование неверного типа данных. Если одна и та же ошибка встречается несколько раз, она считается за одну ошибку	4
Не выполнены условия, позволяющие поставить 4 балла. Программа в целом работает правильно для любых входных данных произвольного размера. Время работы пропорционально количеству введённых чисел; правильно указано, какие величины должны вычисляться по ходу чтения	3

элементов последовательности чисел. Количество синтаксических ошибок («описок») указанных выше видов – не более пяти. Используемая память, возможно, зависит от количества прочитанных чисел (например, входные данные запоминаются в массиве, контейнере STL в C++ или другой структуре данных). Допускается ошибка при вводе и выводе данных, не влияющая на содержание решения. Программа может содержать не более пяти синтаксических и приравненных к ним ошибок, описанных в критериях на 4 балла. Кроме того, допускается наличие одной ошибки, принадлежащей к одному из следующих видов: 1) ошибка инициализации, в том числе отсутствие инициализации; 2) не выводится результат, равный 0, или вместо 0 выводится неверное значение; 3) допущен выход за границу массива; 4) используется знак “<” вместо “<=”, “or” вместо “and” и т.п.	
Не выполнены условия, позволяющие поставить 3 или 4 балла. Программа работает в целом верно, эффективно или нет, например для решения задачи используется перебор всех возможных вариантов выбора элементов в парах. В реализации алгоритма допускается до трёх содержательных ошибок, допустимые виды ошибок перечислены в критериях на 3 балла. Количество синтаксических «описок» не должно быть более семи. Программа может быть неэффективна по времени, например все числа запоминаются в массиве и перебираются все возможные суммы, т.е., по сути, реализовано решение задачи А без ограничений на количество введённых пар	2
Не выполнены условия, позволяющие поставить 2, 3 или 4 балла. Из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи. Допускается любое количество «описок». 1 балл ставится также за решения, верные лишь в частных случаях	1
Не выполнены критерии, позволяющие поставить 1, 2, 3 или 4 балла	0
<i>Максимальный балл для задания Б</i>	4
<i>Итоговый максимальный балл</i>	4

