

## Решения

### ↑ Задание 1 № 6405 тип 1

Переведите в шестнадцатеричную систему счисления двоичное число 101011.

#### Пояснение.

Переведем число в десятичную систему счисления:

$$101011 = 1 \cdot 2^5 + 1 \cdot 2^3 + 1 \cdot 2^1 + 1 \cdot 2^0 = 32 + 8 + 2 + 1 = 43.$$

Десятичное число 43 в шестнадцатеричной системе счисления записывается как 2В.

Ответ: 2В.

Ваш ответ: *нет ответа*. Правильный ответ: 2В

### ↑ Задание 2 № 10466 тип 2

Каждое из логических выражений F и G содержит 5 переменных. В таблицах истинности выражений F и G есть ровно 5 одинаковых строк, причём ровно в 4 из них в столбце значений стоит 1. Сколько строк таблицы истинности для выражения  $F \vee G$  содержит 1 в столбце значений?

#### Пояснение.

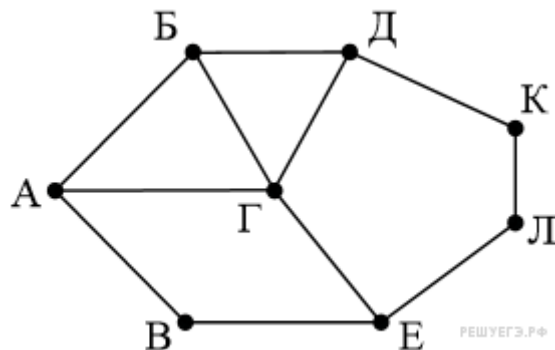
На 4 наборах входных переменных оба выражения равны 1, на 1 наборе оба равны 0, а на всех остальных одно из них равно 0, а другое 1. Поэтому если взять *логическое или* от этих двух выражений, то на том наборе, на котором они оба были равны 0, полученное выражение будет равно 0, на всех же остальных наборах хотя бы одно из них будет равно 1, поэтому и итоговое выражение будет равно 1. Всего различных наборов 32, из них на одном 0, то есть на 31 оставшихся наборах будет 1.

Ваш ответ: *нет ответа*. Правильный ответ: 31

### ↑ Задание 3 № 13560 тип 3

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

	п1	п2	п3	п4	п5	п6	п7	п8
п1		15		20				18
п2	15		25					
п3		25				24		22
п4	20						12	
п5						13	16	17
п6			24		13			15
п7				12	16			
п8	18		22		17	15		



Так как таблицу и схему рисовали независимо друг от друга, нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите длину дороги из пункта Е в пункт Л. В ответе запишите целое число.

#### Пояснение.

Сопоставим таблицу и схему дорог.

Из пункта Г ведёт четыре дороги, только из пункта П8 ведёт четыре дороги, значит, П8 и есть пункт Г.

Из пункта В ведёт две дороги, каждая в пункт, имеющий три дороги. Из пункта П2 также ведут две дороги в пункты с тремя дорогами, значит, пункт П2 и есть пункт В.

Из пункта Б ведёт три дороги, одна — в пункт с тремя дорогами, вторая — в пункт с четырьмя дорогами, третья — в пункт с тремя дорогами. Из пункта П6 дороги идут аналогично. Значит, пункт П6 и есть пункт Б.

Из пункта А ведут дороги в пункты Б, Г и В. Видим из таблицы, что из пункта П3 дороги ведут в Б, Г и В, значит, П3 и есть пункт А.

Из пункта Д ведут дороги в Б, Г и К. Из таблицы видим, что из пунктов П3 и П5 идут дороги в Б и Г, известно, что пункт П3 — А, значит, П5 и есть пункт Д.

Из пункта К ведут две дороги, одна — в пункт Д, другая — в пункт Л, имеющий две дороги. Из таблицы видим, что аналогичные пути идут из пункта П7, значит, П7 и есть пункт К, и П4 — пункт Л.

Из пункта Г ведут четыре дороги — в А, Б, Д и Е, известно, что А — пункт П3, Б — пункт П6, Д — пункт П5, следовательно, П1 и есть пункт Е.

Из таблицы найдём, что длина дороги из пункта Е в пункт Л равна 20.

Ответ: 20.

Ваш ответ: *нет ответа*. Правильный ответ: 20

#### ↑ Задание 4 № 1406 тип 4

Ниже в табличной форме представлен фрагмент базы данных производителей молочных изделий. В первой таблице отражены названия фирм-производителей и торговых точек, с которыми они сотрудничают, во второй — названия фирм-производителей, мест расположения производственных цехов и фамилии ответственных за поставку товаров менеджеров.

Производитель	Торговая точка	Производитель	База производства	Менеджер
Моя бурёнка	ТЦ «Эдельвейс»	Моя буренка	Московская обл.	Иванова И. И.
Корова Му-Му	Рынок Центральный	Корова Му-Му	Московская обл.	Ручкин А. А.
Из деревни с молоком	Рынок Областной	Из деревни с молоком	Московская обл.	Мешков Р. Г.
Парное молочко	ТЦ «Покупочка»	Парное молочко	Московская обл.	Ким Ю. Б.
Сырное царство	ТЦ «Покупочка»	Сырное царство	Ивановская обл.	Рыбкин Н. Н.
Парное молочко	Рынок Центральный	Сырное царство	Вологодская обл.	Охупкин Р. Р.
Корова Му-Му	Рынок Областной	Сырное царство	Рязанская обл.	Рыбкин Н. Н.
Моя буренка	Рынок Центральный	Моя буренка	Вологодская обл.	Иванова И. И.
Сырное царство	ТЦ «Эдельвейс»	Корова Му-Му	Ивановская обл.	Петрова В. Г.

Руководствуясь приведенными таблицами, определите максимальное количество областей, молочные товары которых могут попасть на прилавки Центрального рынка.

#### Пояснение.

1. Определим из первой таблицы производителей, сотрудничающих с Центральным рынком: "Корова Му-Му, Парное молочко, Моя буренка"

2. Определим из второй таблицы, из какой области каждый производитель:

Производитель	База производства	Менеджер
Моя буренка	Московская обл.	Иванова И. И.
Корова Му-Му	Московская обл.	Ручкин А. А.

Из деревни с молоком	Московская обл.	Мешков Р. Г.
Парное молочко	Московская обл.	Ким Ю. Б.
Сырное царство	Ивановская обл.	Рыбкин Н. Н.
Сырное царство	Вологодская обл.	Охапкин Р. Р.
Сырное царство	Рязанская обл.	Рыбкин Н. Н.
Моя буренка	Вологодская обл.	Иванова И. И.
Корова Му-Му	Ивановская обл.	Петрова В. Г.

Следовательно всего 3 различных области, товары из которых могут попасть на прилавки Центрального рынка.

Ответ: 3.

Ваш ответ: *нет ответа*. Правильный ответ: 3

### ↑ Задание 5 № 13732 тип 5

По каналу связи передаются шифрованные сообщения, содержащие только десять букв: А, Б, Е, И, К, Л, Р, С, Т, У. Для передачи используется неравномерный двоичный код. Для девяти букв используются кодовые слова. Для буквы А — 00, Е — 010, И — 011, К — 1111, Л — 1101, Р — 1010, С — 1110, Т — 1011, У — 100.

Укажите кратчайшее кодовое слово для буквы Б, при котором код будет удовлетворять условию Фано. Если таких кодов несколько, укажите код с наименьшим числовым значением.

*Примечание.* Условие Фано означает, что никакое кодовое слово не является началом другого кодового слова. Это обеспечивает возможность однозначной расшифровки закодированных сообщений.

#### Пояснение.

Перечислим возможные коды (не использующиеся для кодировки других букв) в порядке возрастания длины и числового значения.

- 0 — нельзя, А, Е, И начинаются с 0.
  - 1 — нельзя, буквы К, Л, Р, С, Т, У начинаются с 1.
  - 01 — нельзя из-за Е и К.
  - 10 — нельзя из-за Р, Т и У.
  - 11 — нельзя из-за К, Л, С.
  - 000 — нельзя из-за А.
  - 001 — нельзя из-за А.
  - 101 — нельзя из-за Р и Т.
  - 110 — нельзя из-за Л.
  - 111 — нельзя из-за К.
  - 1000 — нельзя из-за У.
  - 1001 — нельзя из-за У.
  - 1100 — можно использовать.
- Таким образом, кратчайшее кодовое слово для буквы Б — 1100.

Ответ: 1100.

Ваш ответ: *нет ответа*. Правильный ответ: 1100

### ↑ Задание 6 № 13733 тип 6

На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему новое число R следующим образом.

- 1) Строится двоичная запись числа N.
- 2) К этой записи дописываются справа ещё два разряда по следующему правилу:
  - а) складываются все цифры двоичной записи числа N, и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
  - б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы её цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R.

Укажите минимальное число  $R$ , которое превышает число 83 и может являться результатом работы данного алгоритма. В ответе это число запишите в десятичной системе счисления.

**Пояснение.**

Рассмотрим числа, большие 83, и найдем меньшее число, которое является результатом работы алгоритма.

$84 = 1010100_2$  — не является результатом работы алгоритма.

$85 = 1010101_2$  — не является результатом работы алгоритма.

$86 = 1010110_2$  — является результатом работы алгоритма для числа  $10101_2$ .

Таким образом, искомое число — 86.

Ответ: 86.

Ваш ответ: *нет ответа*. Правильный ответ: 86

↑ **Задание 7 № 13353 тип 7**

Дан фрагмент электронной таблицы.

	<b>A</b>	<b>B</b>	<b>C</b>
1		3	10
2	$=(A1-3)/(B1+3)$	$=(A1-2)/(C1-3)$	$= C1/(A1 - 4)$



Какое целое число должно быть записано в ячейке A1, чтобы диаграмма, построенная по значениям ячеек диапазона A2:C2, соответствовала рисунку? Известно, что все значения ячеек из рассматриваемого диапазона неотрицательны.

**Пояснение.**

Подставим значения B1 и C1 в формулы A2:C2:

$$A2 = (A1-3)/6$$

$$B2 = (A1-2)/7$$

$$C2 = 10/(A1-4)$$

Так как  $A2 = B2$ , то получим:  $7A1 - 21 = 6A1 - 12$ , откуда  $A1 = 9$ .

Ответ: 9.

Ваш ответ: *нет ответа*. Правильный ответ: 9

↑ **Задание 8 № 3248 тип 8**

Определите, что будет напечатано в результате работы следующего фрагмента программы:

<b>Бейсик</b>	<b>Python</b>
<pre> DIM N, S AS INTEGER N = 12 S = 5 WHILE N &lt;= 25     S = S + 12     N = N + 2 WEND PRINT S </pre>	<pre> n = 12 s = 5 while n &lt;= 25:     s += 12     n += 2 print(s) </pre>
<b>Паскаль</b>	<b>Алгоритмический язык</b>
<pre> var n, s: integer; begin     n := 12;     s := 5; </pre>	<pre> алг нач     цел n, s     n := 12 </pre>

<pre> while n &lt;= 25 do begin   s := s + 12;   n := n + 2; end; writeln(s); end. </pre>	<pre> s := 5 нц пока n &lt;= 25   s := s + 12   n := n + 2 кц вывод s кон </pre>
<b>Си++</b>	
<pre> #include &lt;iostream&gt; using namespace std; int main() {   int n, s;   n = 12, s = 5;   while (n &lt;= 25) {     s = s + 12;     n = n + 2;   }   cout &lt;&lt; s &lt;&lt; endl;   return 0; } </pre>	

**Пояснение.**

Цикл while выполняется до тех пор, пока истинно условие  $n \leq 25$ , т. е. переменная  $n$  определяет, сколько раз выполнится цикл.

Аккуратно посчитаем, сколько раз выполниться цикл ( $k$ ):

Значения  $n$ : 12 14 16 18 20 22 24 26

Следовательно, цикл выполнится семь раз (так как действие  $n := n + 2$  стоит в конце цикла, следовательно, для  $n=26$  действие  $s := s + 12$  не будет выполняться).

Посчитаем значение  $s$ :  $s = 12 \cdot k + s_1 = 12 \cdot 7 + 5 = 89$ .

Ваш ответ: *нет ответа*. Правильный ответ: 89

### ↑ Задание 9 № 5314 тип 9

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.

**Пояснение.**

Вычислим скорость передачи данных по каналу связи:  $\frac{24\text{Мбайт}}{80\text{сек}} = \frac{3}{10}\text{Мбайт/сек}$ .

Поскольку на упаковку и распаковку данных ушло 10 секунд, то на передачу сжатых данных по каналу связи ушло  $20 - 10 = 10$  секунд.

Зная скорость передачи, найдем объем сжатого файла:  $10\text{сек} \cdot \frac{3}{10}\text{Мбайт/сек} = 3\text{Мбайт}$ .

Ответ: 3 Мбайт.

Ваш ответ: *нет ответа*. Правильный ответ: 3

### ↑ Задание 10 № 3208 тип 10

Все 5-буквенные слова, составленные из букв К, О, Р, записаны в алфавитном порядке и пронумерованы. Вот начало списка:

1. ККККК
2. ККККО
3. ККККР
4. КККОК

.....

Запишите слово, которое стоит под номером 182.

#### Пояснение.

Заменяем буквы К, О, Р на 0, 1, 2 (для них порядок очевиден – по возрастанию).

Выпишем начало списка, заменив буквы на цифры:

1. 00000
2. 00001
3. 00002
4. 00010

...

Полученная запись есть числа, записанные в троичной системе счисления в порядке возрастания. Тогда на 182 месте будет стоять число 181 (т. к. первое число 0). Переведём число 181 в троичную систему (деля и снося остаток справа налево):

$$\begin{aligned} 181 / 3 &= 60 (1) \\ 60 / 3 &= 20 (0) \\ 20 / 3 &= 6 (2) \\ 6 / 3 &= 2 (0) \\ 2 / 3 &= 0(2) \end{aligned}$$

В троичной системе 181 запишется как 20201. Произведём обратную замену и получим РКРКО.

Ответ: РКРКО.

Ваш ответ: *нет ответа*. Правильный ответ: РКРКО

### ↑ Задание 11 № 15796 тип 11

Ниже на пяти языках программирования записан рекурсивный алгоритм F.

Бейсик	Python
<pre> SUB F(n)   IF n &gt; 0 THEN     F(n - 3)     PRINT N     F(n \ 3)   END IF END SUB </pre>	<pre> def F(n):   if n &gt; 0:     F(n - 3)     print(n)     F(n // 3)   else: return n </pre>
Паскаль	Алгоритмический язык
<pre> procedure F(n: integer); begin   if n &gt; 0 then begin     F(n - 3);     write(n);     F(n div 3);   end end; </pre>	<pre> алг F(цел n) нач   если n &gt; 0 то     F(n - 3)     вывод n     F(div(n,3))   все кон </pre>

<b>C++</b>
<pre> void F (int n) {     if (n &gt; 0) {         F (n - 3);         std::cout &lt;&lt; n;         F (n / 3);     } } </pre>

Запишите подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова F(9). Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

**Пояснение.**

Первым действием процедура F(9) вызовет процедуру F(n-3), т.е. F(6), которая вызовет процедуру F(3), которая вызовет процедуру F(0) и следующим шагом своего алгоритма выведет на экран число 3, после чего обратится к процедуре F(n/3), т.е. вызовет процедуру F(1), после чего на экран будет выведена единица. После этого управление вернётся к процедуре F(6), которая начнёт выполнять следующий шаг своего алгоритма и выведет на экран число 6. После этого будет вызвана процедура F(2), после чего на экран будет выведено число 2.

После этого управление вернётся к процедуре F(9), которая следующим шагом своего алгоритма выведет на экран число 9, после чего обратится к процедуре F(n/3), т.е. обратится к процедуре F(3), которая вызовет процедуру F(0), после чего на экран будет выведено число 3. После этого будет вызвана процедура F(1), и на экран будет выведена единица.

Ответ: 3162931.

Ваш ответ: *нет ответа*. Правильный ответ: 3162931

↑ **Задание 12 № 5879 тип 12**

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

IP-адрес узла: 129.131.130.64  
 Маска: 255.255.192.0

При записи ответа выберите из приведённых в таблице чисел четыре элемента IP-адреса сети и запишите в нужном порядке соответствующие им буквы без использования точек.

A	B	C	D	E	F	G	H
0	64	128	129	130	131	192	255

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

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

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

**Пояснение.**

1. Запишем числа маски сети в двоичной системе счисления.

$$255_{10} = 11111111_2$$

$$192_{10} = 11000000_2$$

$$0_{10} = 00000000_2$$

2. Адрес сети получается в результате поразрядной конъюнкции чисел маски и чисел адреса узла (в двоичном коде). Так как конъюнкция 0 с чем-либо всегда равна 0, то на тех местах, где числа маски равны 0, в адресе узла стоит 0. Аналогично, там, где числа маски равны 255, стоит само число, так как конъюнкция 1 с любым числом всегда равна этому числу.

3. Рассмотрим конъюнкцию числа 192 с числом 130.

$$\begin{aligned} 192_{10} &= 11000000_2 \\ 130_{10} &= 1000010_2 \end{aligned}$$

Результатом конъюнкции является число  $10000000_2 = 128$ .

4. Сопоставим варианты ответа получившимся числам: 129, 131, 128, 0.

Таким образом, ответ: DFCA.

Ваш ответ: *нет ответа*. Правильный ответ: DFCA

### ↑ Задание 13 № 212 тип 13

В одной небольшой стране Индивидуальные Номера Налогоплательщиков представляют собой целые числа от 1 до 4000, На некотором предприятии в этой стране работают 300 человек. Главный бухгалтер этого предприятия переписала ИНН всех сотрудников последовательно без разделителей в один файл, при этом использовалось представление целых чисел с одинаковым минимально возможным количеством бит. Оцените объем получившегося файла. (Ответ дайте в байтах.)

#### Пояснение.

Известно, что с помощью  $N$  бит можно закодировать  $2^N$  различных чисел. Поскольку  $2^{11} < 4000 < 2^{12}$ , то для записи каждого из 4000 ИНН необходимо 12 бит памяти. Поэтому 300 номеров, записанных подряд займут  $300 \cdot 12 = 3600$  бит = 450 байт.

Ваш ответ: *нет ответа*. Правильный ответ: 450

### ↑ Задание 14 № 7759 тип 14

Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **сместиться на (a, b)**, где  $a, b$  – целые числа. Эта команда перемещает Чертёжника из точки с координатами  $(x, y)$  в точку с координатами  $(x + a, y + b)$ . Например, если Чертёжник находится в точке с координатами  $(4, 2)$ , то команда сместиться на  $(2, -3)$  переместит Чертёжника в точку  $(6, -1)$ .

Цикл

ПОВТОРИ число РАЗ

последовательность команд

КОНЕЦ ПОВТОРИ

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

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

НАЧАЛО

**сместиться на (30, -10)**

ПОВТОРИ  $n$  РАЗ

**сместиться на (a, b)**

**сместиться на (-11, -12)**

КОНЕЦ ПОВТОРИ

**сместиться на (-3, 100)**

КОНЕЦ

Укажите наименьшее возможное значение числа  $n$ , для которого найдутся такие значения чисел  $a$  и  $b$ , что после выполнения программы Чертёжник возвратится в исходную точку.



**Пояснение.**

После выполнения команд сместиться на  $(30, -10)$  и сместиться на  $(-3, 100)$  Чертёжник окажется в точке с координатами  $(27, 90)$ . После выполнения цикла Чертёжник переместится на  $n \cdot (a - 11, b - 12)$ .

Поскольку требуется, чтобы после выполнения программы Чертёжник вернулся в исходную точку, имеем два уравнения:  $n \cdot (a - 11) = -27$  и  $n \cdot (b - 12) = -90$ .

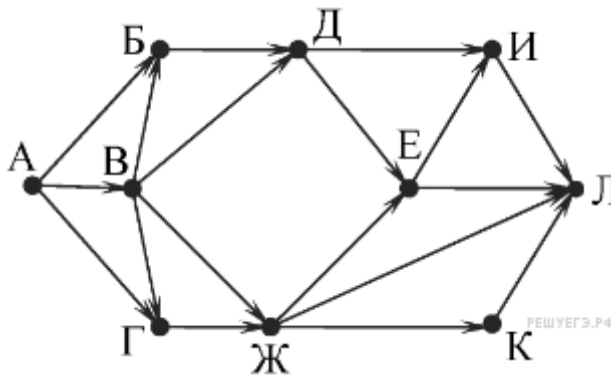
Переменные  $a$ ,  $b$  и  $n$  должны быть целыми, причём  $n > 1$ . Следовательно, числа  $-90$  и  $-27$  должны быть кратны  $n$ . Наименьшее, подходящее  $n$  равно 3.

Ответ: 3.

Ваш ответ: *нет ответа*. Правильный ответ: 3

↑ **Задание 15 № 5781 тип 15**

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

**Пояснение.**

Начнем считать количество путей с конца маршрута — с города Л.  $N_X$  — количество различных путей из города А в город X,  $N$  — общее число путей.

В "Л" можно приехать из И, Е, Ж или К, поэтому  $N = N_L = N_I + N_E + N_{Zh} + N_K$  (\*)

Аналогично:

$$\begin{aligned} N_I &= N_D + N_E = 3 + 6 = 9; \\ N_E &= N_D + N_{Zh} = 3 + 3 = 6; \\ N_{Zh} &= N_V + N_G = 1 + 2 = 3; \\ N_K &= N_{Zh} = 3. \end{aligned}$$

Добавим еще вершины:

$$\begin{aligned} N_D &= N_B + N_V = 2 + 1 = 3; \\ N_B &= N_A = 1; \\ N_G &= N_A + N_V = 1 + 1 = 2; \\ N_V &= N_A + N_B = 1 + 1 = 2. \end{aligned}$$

Подставим в формулу (\*):

$$N = N_L = 9 + 6 + 3 + 3 = 21.$$

Ваш ответ: *нет ответа*. Правильный ответ: 21

↑ **Задание 16 № 14231 тип 16**

В системе счисления с основанием  $N$  запись числа  $144_{10}$  оканчивается на 1 и содержит не менее трёх цифр. Чему равно число  $N$ ?

**Пояснение.**

Пусть системе счисления с основанием  $N$  число записано в виде  $xy1$ . Тогда  $144 = x \cdot N^2 + y \cdot N + 1$  или  $N(xN + y) = 11 \cdot 13$ . Поскольку  $x$  — натуральное число, выражение в скобке больше  $N$  (\*).

Пусть  $N = 11$ ,  $xN + y = 13$ . Тогда  $x = 1$ ,  $y = 2$ . Если  $x > 1$ , то сумма в скобке больше  $13$ :  $x \cdot 11 + y > 13$ , поэтому других решений нет.

Второй случай:  $N = 13$ ,  $xN + y = 11$ . Он невозможен в силу наблюдения (\*).

Ответ: 11.

Ваш ответ: *нет ответа*. Правильный ответ: 11

↑ **Задание 17 № 6465 тип 17**

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 тысячах)
Бюффон & (Руссо   Вольтер)	460
Бюффон & Руссо & Вольтер	110
Бюффон & Руссо	260

Компьютер печатает количество страниц (в тысячах), которое будет найдено по следующему запросу:

*Бюффон & Вольтер.*

Укажите целое число, которое напечатает компьютер. Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

**Пояснение.**

Количество запросов в данной области будем обозначать  $N_i$ . Наша цель —  $N_4 + N_5$ . Тогда из таблицы находим, что:

$$N_2 + N_5 = 260,$$

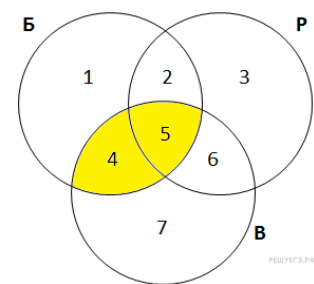
$$N_5 = 110,$$

$$N_2 + N_4 + N_5 = 460.$$

Из первого и второго равенств находим:  $N_2 = 150$ , из последнего равенства:  $N_4 + N_5 = 310$ .

Ответ: 310.

Ваш ответ: *нет ответа*. Правильный ответ: 310



↑ **Задание 18 № 10294 тип 18**

Элементами множеств  $A$ ,  $P$ ,  $Q$  являются натуральные числа, причём  $P = \{1, 3, 4, 9, 11, 13, 15, 17, 19, 21\}$ ,  $Q = \{3, 6, 9, 12, 15, 18, 21, 24, 27, 30\}$ .

Известно, что выражение

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

истинно (то есть принимает значение 1) при любом значении переменной  $x$ . Определите наименьшее возможное количество элементов в множестве  $A$ .

**Пояснение.**

Упростим выражение:

$$\begin{aligned} & ((x \in P) \rightarrow (x \in A)) \vee (\neg(x \in A) \rightarrow \neg(x \in Q)) \\ & ((x \notin P) \vee (x \in A)) \vee ((x \in A) \vee (x \notin Q)) \\ & (x \notin P) \vee (x \in A) \vee (x \notin Q) \end{aligned}$$

Таким образом, выражение верно для тех  $x$ , которые либо не принадлежат  $P$ , либо не принадлежат  $Q$ , либо принадлежат  $A$ . То есть сейчас выражение ложно только для тех  $x$ , которые принадлежат одновременно и  $P$ , и  $Q$ . Это 3, 9, 15 и 21. Если эти числа положить в  $A$ , то тогда выражение будет верно всегда.

Ваш ответ: *нет ответа*. Правильный ответ: 4

↑ **Задание 19 № 11275 тип 19**

В программе используется одномерный целочисленный массив  $A$  с индексами от 0 до 10. Значения элементов равны 4, 3, 6, 8, 4, 8, 2, 10, 9, 14, 4 соответственно, т. е.  $A[0] = 4$ ,  $A[1] = 3$  и т. д. Определите значение переменной  $c$  после выполнения следующего фрагмента программы (записанного ниже на пяти языках программирования).

Бейсик	Python
<pre>c = 0 FOR i = 1 TO 9   IF A(i) = A(0) THEN     c = c + 1     t = A(i+1)     A(i+1) = A(i)     A(i) = t   END IF NEXT i</pre>	<pre>c = 0 for i in range(1,10):   if A[i] == A[0]:     c = c + 1     t = A[i+1]     A[i+1] = A[i]     A[i] = t</pre>
Алгоритмический язык	Паскаль
<pre>c := 0 нц для i от 1 до 9   если A[i] = A[0] то     c := c + 1     t := A[i+1]     A[i+1] := A[i]     A[i] := t   все кц</pre>	<pre>c := 0; for i := 1 to 9 do begin   if A[i] = A[0] then begin     c := c + 1;     t := A[i+1];     A[i+1] := A[i];     A[i] := t;   end; end;</pre>
Си++	
<pre>c = 0; for (i = 1; i &lt; 10; i++) {   if (A[i] == A[0]) {     c++;     t = A[i+1];     A[i+1] = A[i];     A[i] = t;   } }</pre>	

**Пояснение.**

Заметим, что первый раз условие в цикле выполнится, когда  $A[i] = A[0]$ . После этого значения  $A[i]$  и  $A[i+1]$  поменяются местами, поэтому на всех последующих шагах условие  $A[i] = A[0]$  также будет

выполнено. Каждый раз при выполнении условия  $A[i] = A[0]$  переменная  $c$  увеличивается на единицу. Первый раз условие будет выполнено при  $i=4$ , поэтому после выполнения фрагмента программы  $c$  будет равно 6.

Ответ: 6.

Ваш ответ: *нет ответа*. Правильный ответ: 6

### ↑ Задание 20 № 5940 тип 20

Ниже на пяти языках записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа:  $a$  и  $b$ . Укажите наименьшее из таких чисел  $x$ , при вводе которых алгоритм печатает сначала 15, а потом 6.

Бейсик	Паскаль
<pre> DIM X, A, B, C AS INTEGER INPUT X A = 0: B = 10 WHILE X &gt; 0     C = X MOD 10     A = A + C     IF C &lt; B THEN B = C     X = X \ 10 WEND PRINT A PRINT B </pre>	<pre> var x, a, b, c: integer; begin     readln(x);     a := 0; b := 10;     while x&gt;0 do         begin             c := x mod 10;             a := a+c;             if c &lt; b then b := c;             x := x div 10;         end;     writeln(a); write(b); end. </pre>
Си++	Алгоритмический
<pre> #include &lt;iostream&gt; using namespace std; int main() {     int x, a, b, c;     cin &gt;&gt; x ;     a = 0; b = 10;     while (x&gt;0) {         c = x%10;         a = a+c;         if (c &lt; b)             b = c;         x = x /10;     }     cout &lt;&lt; a &lt;&lt; endl &lt;&lt; b endl; } </pre>	<pre> алг нач цел x, a, b, c ввод x a := 0; b := 10 нц пока x&gt;0     c := mod(x,10)     a := a+c     если c&lt;b         то b := c     все     x := div(x,10) кц вывод a, нe, b кон </pre>
Python	
<pre> x = int(input()) a = 0 b = 10 while x &gt; 0:     c = x % 10     a += c     if c &lt; b:         b = c     x //= 10 print(a) print(b) </pre>	

**Пояснение.**

Рассмотрим цикл, число шагов которого зависит от изменения переменной  $x$ :

```
while x > 0 do begin
...
x:= x div 10;
end;
```

Т. к. оператор `div` оставляет только целую часть от деления, то при делении на 10 это равносильно отсечению последней цифры.

Из приведенного цикла видно, что на каждом шаге от десятичной записи  $x$  отсекается последняя цифра до тех пор, пока все цифры не будут отсечены, то есть  $x$  не станет равно 0; поэтому цикл выполняется столько раз, сколько цифр в десятичной записи введенного числа.

При этом, переменной  $s$  присваивается остаток от деления  $x$  на 10, переменная  $a$  увеличивается на остаток от деления  $x$  на 10. Если остаток от деления  $x$  на 10 меньше переменной  $b$ , то переменной  $b$  присваивается значение переменной  $s$ . Поскольку требуется, чтобы программа напечатала сначала 15, необходимо, чтобы сумма цифр числа  $x$  была равна 15. Поскольку вторым должно быть напечатано число 6, необходимо, чтобы первой цифрой числа  $x$  была цифра 6. Наименьшее такое число — 69.

Ответ: 69.

Ваш ответ: *нет ответа*. Правильный ответ: 69

↑ **Задание 21 № 14782 тип 21**

Напишите в ответе **наибольшее** значение входной переменной  $k$ , при котором программа выдаёт ответ 9. Для Вашего удобства программа приведена на пяти языках программирования.

Бейсик	Python
<pre>DIM K, I AS LONG INPUT K I = 20 WHILE F(I) &gt; K     I = I - 1 WEND PRINT I FUNCTION F(N)     F = N * N * N END FUNCTION</pre>	<pre>def f(n):     return n * n * n k = int(input()) i = 20 while f(i) &gt; k:     i -= 1 print(i)</pre>
Паскаль	Алгоритмический язык
<pre>var     k, i : longint; function f(n: longint): longint; begin     f := n * n * n end; begin     readln(k);     i := 20;     while f(i) &gt; k do         i := i-1;     writeln(i) end.</pre>	<pre>алг нач     цел k, i     ввод k     i := 20     нц пока f(i) &gt; k         i := i - 1     кц     вывод i кон алг цел f(цел n) нач     знач := n * n * n кон</pre>
Си++	
<pre>#include &lt;iostream&gt; using namespace std; long f(long n) { return n * n * n; }</pre>	

```

int main()
{
    long k, i;
    cin >> k;
    i = 20;
    while (f(i) > k) --i;
    cout << i;
    return 0;
}

```

**Пояснение.**

На печать выводится значение  $i$ , уменьшающееся при каждом выполнении цикла на единицу, идя от 20 до 9. Следовательно, алгоритм выполнится ровно 11 раз. Поэтому, условие  $x^3 > k$  должно быть истинным первые 11 раз. Тогда  $9^3 \leq k < 10^3$ . Тем самым, наибольшее возможное значение равно 999.

Ответ: 999.

Ваш ответ: *нет ответа*. Правильный ответ: 999

### ↑ Задание 22 № 3791 тип 22

У исполнителя Увеличитель две команды, которым присвоены номера:

1. прибавь 1,
2. умножь на 4.

Первая из них увеличивает число на экране на 1, вторая — умножает его на 4.

Программа для Увеличителя — это последовательность команд. Сколько есть программ, которые число 1 преобразуют в число 32?

Ответ обоснуйте.

**Пояснение.**

Обозначим  $R(n)$  — количество программ, которые преобразуют число 1 в число  $n$ . Обозначим  $t(n)$  наибольшее число, кратное четырем, не превосходящее  $n$ .

Обе команды исполнителя увеличивают исходное число, поэтому общее количество команд в программе не может превосходить 32.

Верны следующие соотношения:

1. Если  $n$  не делится на 4, то тогда  $R(n) = R(t(n))$ , так как существует единственный способ получения  $n$  из  $t(n)$  - прибавлением единиц.

2. Пусть  $n$  делится на 4.

Тогда  $R(n) = R(n/4) + R(n-1) = R(n/4) + R(n-4)$  (если  $n > 4$ ).

При  $n = 4$  выполнено:  $R(n) = 2$  (два способа: прибавлением трех единиц или однократным умножением на 4).

Поэтому достаточно по индукции вычислить значения  $R(n)$  для всех нечётных чисел, кратных четырем и не превосходящих 32.

Имеем:

$$R(1) = R(2) = R(3) = 1$$

$$R(4) = 2 = R(5) = R(6) = R(7)$$

$$R(8) = R(2) + R(7) = 1 + 2 = 3 = R(9) = R(10) = R(11)$$

$$R(12) = R(3) + R(11) = 1 + 3 = 4 = R(13) = R(14) = R(15)$$

$$R(16) = R(4) + R(15) = 2 + 4 = 6 = R(17) = R(18) = R(19)$$

$$R(20) = R(5) + R(19) = 2 + 6 = 8 = R(21) = R(22) = R(23)$$

$$R(24) = R(6) + R(23) = 2 + 8 = 10 = R(25) = R(26) = R(27)$$

$$R(28) = R(7) + R(27) = 2 + 10 = 12 = R(29) = R(30) = R(31)$$

$$R(32) = R(8) + R(31) = 3 + 12 = 15$$

Ответ: 15.

Другой способ решения

Будем решать поставленную задачу последовательно для чисел 1, 2, 3, ..., 32 (то есть для каждого из чисел определим, сколько программ исполнителя существует для его получения). Количество программ, которые преобразуют число 1 в число n, будем обозначать через R(n). Число 1 у нас уже есть, значит, его можно получить с помощью «пустой» программы. Любая непустая программа увеличит исходное число, т. е. даст число, больше 1. Значит,  $R(1) = 1$ . Для каждого следующего числа рассмотрим, из какого числа оно может быть получено за одну команду исполнителя. Если число не делится на 4, то оно может быть получено только из предыдущего числа с помощью команды **прибавь 1**. Значит, количество искомых программ для такого числа равно количеству программ для предыдущего числа:  $R(i) = R(i - 1)$ .

Если число на 4 делится, то вариантов последней команды два: **прибавь 1 и умножь на 4**, тогда  $R(i) = R(i - 1) + R(i/4)$ . Заполним соответствующую таблицу по приведённым формулам слева направо:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	6
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
6	6	6	8	8	8	8	10	10	10	10	12	12	12	12	15

При этом ячейки, относящиеся к числам, которые не делятся на 4, можно в решении и опустить (за исключением первого и последнего чисел):

1	4	8	12	16	20	24	28	32
1	2	3	4	6	8	10	12	15

Ответ: 15.

Ваш ответ: *нет ответа*. Правильный ответ: 15

↑ **Задание 23 № 9704 тип 23**

Сколько существует различных наборов значений логических переменных  $x_1, x_2, \dots, x_{10}$ , которые удовлетворяют всем перечисленным ниже условиям:

$$\begin{aligned} &\neg((x_1 \wedge \neg x_2 \wedge x_3) \vee (\neg x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \neg x_2 \wedge \neg x_3)) = 1, \\ &\neg((x_2 \wedge \neg x_3 \wedge x_4) \vee (\neg x_2 \wedge x_3 \wedge x_4) \vee (x_2 \wedge \neg x_3 \wedge \neg x_4)) = 1, \\ &\dots \\ &\neg((x_8 \wedge \neg x_9 \wedge x_{10}) \vee (\neg x_8 \wedge x_9 \wedge x_{10}) \vee (x_8 \wedge \neg x_9 \wedge \neg x_{10})) = 1. \end{aligned}$$

В ответе не нужно перечислять все различные наборы значений переменных  $x_1, x_2, \dots, x_{10}$ , при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

**Пояснение.**

Преобразуем уравнение:

$$\begin{aligned} &\neg((x_n \wedge \neg x_{n+1} \wedge x_{n+2}) \vee (\neg x_n \wedge x_{n+1} \wedge x_{n+2}) \vee (x_n \wedge \neg x_{n+1} \wedge \neg x_{n+2})) = 1 \\ &\neg(x_n \wedge \neg x_{n+1} \wedge x_{n+2}) \wedge \neg(\neg x_n \wedge x_{n+1} \wedge x_{n+2}) \wedge \neg(x_n \wedge \neg x_{n+1} \wedge \neg x_{n+2}) = 1 \\ &(\neg x_n \vee x_{n+1} \vee \neg x_{n+2}) \wedge (x_n \vee \neg x_{n+1} \vee \neg x_{n+2}) \wedge (\neg x_n \vee x_{n+1} \vee x_{n+2}) = 1. \end{aligned}$$

Данное уравнение принимает значение ложь при таких наборах переменных:

$$\left[ \begin{aligned} &x_n = 1, x_{n+1} = 0, x_{n+2} = 1, \\ &x_n = 0, x_{n+1} = 1, x_{n+2} = 1, \\ &x_n = 1, x_{n+1} = 0, x_{n+2} = 0. \end{aligned} \right.$$

Таким образом можем выписать все наборы переменных, удовлетворяющие данному условию  $x_1,$

$x_2, \dots, x_{10}$ :

```
1111111111
1111111110
0000000010
0000000001
0000000000
```

Всего 5 наборов.

Ответ: 5.

Ваш ответ: *нет ответа*. Правильный ответ: 5



## Проверка части с развернутым ответом

Пожалуйста, оцените решения заданий части с развернутым ответом самостоятельно, руководствуясь указанными критериями.

### Задание 24 (С1) № 7469

Обратите внимание! В задаче требовалось выполнить четыре действия:

- 1) указать, что выведет программа при конкретной входной последовательности;
- 2) указать пример последовательности, при которой программа работает правильно;
- 3) исправить первую ошибку;
- 4) исправить вторую ошибку.

Для проверки правильности выполнения п. 2) нужно формально выполнить исходную (ошибочную) программу с входными данными, которые указал экзаменуемый, и убедиться в том, что результат, выданный программой, будет таким же, как и для правильной программы. Для действий 3) и 4) 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.

Критерии оценивания выполнения задания	Баллы
Выполнены все четыре необходимых действия, и ни одна верная строка не указана в качестве ошибочной	3
Не выполнены условия, позволяющие поставить 3 балла. Имеет место одна из следующих ситуаций: а) выполнены три из четырёх необходимых действий. Ни одна верная строка не указана в качестве ошибочной; б) выполнены все четыре необходимых действия. Указано в качестве ошибочной не более одной верной строки.	2
Не выполнены условия, позволяющие поставить 2 или 3 балла. Выполнены два необходимых действия из четырёх.	1
Не выполнены условия, позволяющие поставить 1, 2 или 3 балла.	0
<i>Максимальный балл</i>	3

На обработку поступает последовательность из четырёх неотрицательных целых чисел (некоторые числа могут быть одинаковыми). Нужно написать программу, которая выводит на экран количество нечётных чисел в исходной последовательности и максимальное нечётное число. Если нечётных чисел нет, требуется на экран вывести «NO». Известно, что вводимые числа не превышают 1000. Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на пяти языках программирования.

Бейсик	Python
<pre> CONST n = 4 count = 0 maximum = 999 FOR I = 1 TO n INPUT x IF x mod 2 &lt;&gt; 0 THEN count = count + 1 IF x &gt; maximum THEN maximum = I END IF END IF NEXT I IF count &gt; 0 THEN PRINT count PRINT maximum ELSE </pre>	<pre> n = 4 count = 0 maximum = 999 for i in range(1, n + 1):     x = int(input())     if x % 2 != 0:         count += 1         if x &gt; maximum:             maximum = i if count &gt; 0:     print(count)     print(maximum) else:     print("NO") </pre>

<code>PRINT "NO" END IF</code>	
<b>Паскаль</b>	<b>Алгоритмический язык</b>
<pre> const n = 4; var i, x: integer; var maximum, count: integer; begin count := 0; maximum := 999; for i := 1 to n do begin read(x); if x mod 2 &lt;&gt; 0 then begin count := count + 1; if x &gt; maximum then maximum := i end end; if count &gt; 0 then begin writeln(count); writeln(maximum); end else writeln('NO') end. </pre>	<pre> алг нач цел n = 4 цел i, x цел maximum, count count := 0 maximum := 999 нц для i от 1 до n ввод x если mod(x, 2) &lt;&gt; 0 то count := count + 1 если x &gt; maximum то maximum := i все все кц если count &gt; 0 то вывод count, нс вывод maximum иначе вывод "NO" все кон </pre>
<b>Си++</b>	
<pre> #include &lt;iostream&gt; using namespace std; int main(void) { const int n = 4; int i, x, maximum, count; count = 0; maximum = 999; for (i = 1; i &lt;= n; i++) { cin &gt;&gt; x; if (x % 2 != 0) { count++; if (x &gt; maximum) maximum = i; } } if (count &gt; 0) { cout &lt;&lt; count; cout &lt;&lt; maximum; } else cout &lt;&lt; "NO\n"; } </pre>	

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе последовательности: 2 9 4 3
2. Приведите пример такой последовательности, содержащей хотя бы одно нечётное число, что, несмотря на ошибки, программа печатает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, т. е. приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования. Обратите внимание, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

### Пояснение.

Решение использует запись программы на Паскале.

1. Программа выведет два числа: 2 и 999.
2. Пример последовательности, содержащей нечётные числа, для которой программа работает правильно: 1 2 3 999.

Замечание. В конце работы программы значение переменной `maximum` всегда равно 999. Соответственно, программа будет работать верно, если в последовательности есть 999. Выведенное количество нечётных чисел будет правильным в любом случае.

3. В программе есть две ошибки.

Первая ошибка: неверная инициализация `maximum`.

Строка с ошибкой:

```
maximum := 999;
```

Верное исправление:

```
maximum := 0;
```

Вместо 0 может быть использовано любое число, меньшее или равное 1.

Вторая ошибка: неверное присваивание при вычислении максимума.

Строка с ошибкой:

```
maximum = i;
```

Верное исправление:

```
maximum = x;
```

### Задание 25 (С2) № 5885

Критерии оценивания выполнения задания	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 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. 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: <ol style="list-style-type: none"> <li>1) не инициализируется или неверно инициализируется переменная <code>MAX</code> (например, присваивается начальное значение, большее или равное 100);</li> <li>2) неточно определяется принадлежность диапазону трёхзначных чисел (например, в сравнении со 100 вместо знака «больше или равно» используется знак «больше»);</li> <li>3) неверно осуществляется проверка делимости на 8;</li> <li>4) на делимость на 8 проверяется не значение элемента, а его индекс;</li> <li>5) в сложном условии вместо логической операции «И» используется логическая операция «ИЛИ»;</li> <li>6) отсутствует вывод ответа, или при выводе ответа не учитывается случай, что требуемого числа в массиве может не быть;</li> <li>7) используется переменная, не объявленная в разделе описания переменных;</li> <li>8) не указано или неверно указано условие завершения цикла;</li> <li>9) индексная переменная в цикле не меняется (например, в цикле <code>while</code>) или меняется неверно;</li> <li>10) неверно расставлены операторные скобки.</li> </ol>	1
Ошибок, перечисленных в п. 1-10, две или больше, или алгоритм сформулирован неверно.	0
<i>Максимальный балл</i>	2

Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от 0 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести максимальное значение среди

трёхзначных элементов массива, не делящихся на 8. Если в исходном массиве нет элемента, значение которого является трёхзначным числом и при этом не кратно 8, то вывести сообщение «Не найдено».

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Бэйсик	Паскаль
<pre>N=20 DIM A(N) AS INTEGER DIM I, J, MAX AS INTEGER FOR I=1 TO N   INPUT A(I) NEXT I ... END</pre>	<pre>const   N=20; var   a: array [1..N] of integer;   i, j, max: integer; begin   for i:=1 to N do     readln (a[i]);   ... end.</pre>
Си++	Алгоритмический язык
<pre>#include &lt;iostream&gt; using namespace std; #define N 20 int main () {   int a[N];   int i, j, max;   for (i=0; i&lt;N; i++)     cin &gt;&gt; a[i];   ... }</pre>	<pre>алг нач   цел N=20   целтаб a[1:N]   цел i, j, max   нц для i от 1 до N     ввод a[i]   кц   ... кон</pre>
Естественный язык	
<p>Объявляем массив A из 20 элементов.          Объявляем целочисленные переменные I, J, MAX.          В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.          ...</p>	
Python	
<pre># допускается также # использовать две # целочисленные переменные j, max a = [] n = 20 for i in range(0, n):   a.append(int(input())) ...</pre>	

В качестве ответа Вам необходимо привести фрагмент программы (или описание алгоритма на естественном языке), который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например, *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 (например, в образце, записанном на естественном языке).

#### Пояснение.

Допускаются иные формулировки ответа, не искажающие его смысла.

На языке Бэйсик	На языке Паскаль
<pre>MAX =99 FOR I = 1 TO N</pre>	<pre>max := 99; for i := 1 to N do</pre>

<pre> IF A(I)&gt;=100 AND A(I)&lt;= 999 AND A(I) MOD 8&lt;&gt;0 AND A(I)&gt;MAX THEN MAX = A(I) END IF NEXT I IF MAX &gt;99 THEN PRINT MAX ELSE PRINT "Не найдено" END IF </pre>	<pre> if (a[i]&gt;=100) and (a[i]&lt;=999) and (a[i] mod 8&lt;&gt;0) and (a[i]&gt;max) then max := a[i]; if max &gt; 99 then writeln (max) else writeln('Не найдено'); </pre>
<b>На языке Си</b>	<b>На алгоритмическом языке</b>
<pre> max = 99; for (i = 0; iif (a[i]&gt;99 &amp;&amp; a[i]&lt;1000 &amp;&amp; a[i]%8!=0 &amp;&amp; a[i]&gt;max) max = a[i] ; if (max&gt;99) cout &lt;&lt; max &lt;&lt; endl; else cout &lt;&lt; "Не найдено"; </pre>	<pre> max := 99 нц для i от 1 до N если a[±]&gt;=100 и a[i]&lt;= 999 и mod(a[i],8)&lt;&gt;0 и a[i]&gt;max то max := a[i] все кц если max &gt; 99 то вывод max иначе вывод "Не найдено" все </pre>
<b>На естественном языке</b>	
<p>Записываем в переменную MAX начальное значение, равное 99. В цикле от первого элемента до двадцатого находим остаток от деления элемента исходного массива на 8. Если значение данного остатка не равно 0 и значение текущего элемента массива больше 99 и меньше 1000, то сравниваем значение текущего элемента массива со значением переменной MAX. Если текущий элемент массива больше MAX, то записываем в MAX значение этого элемента массива. Переходим к следующему элементу. После завершения цикла проверяем значение переменной MAX. Если оно больше 99, то выводим его, иначе выводим сообщение «Не найдено».</p>	

### Задание 26 (С3) № 7683

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. 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 одновременно:

- 1) правильно указаны все позиции, в которых Петя выигрывает первым ходом;
- 2) правильно указана позиция, в которой Ваня выигрывает первым ходом;
- 3) явно указано, что при любом ходе Пети Ваня может получить кучу, которая содержит нужное для выигрыша количество камней, но при этом не указано, каким именно ходом выигрывает Ваня.

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. В этом дереве в каждой позиции, где должен ходить Петя, должны быть 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

Критерии оценивания выполнения задания	Баллы
<p>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.</p>	3
<p>Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий.</p> <ul style="list-style-type: none"> <li>– Третье задание выполнено полностью.</li> <li>– Первое и второе задания выполнены полностью.</li> <li>– Первое задание выполнено полностью или частично, для второго и третьего заданий указаны правильные значения S.</li> </ul>	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

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 **увеличить количество камней в куче в шесть раз**. Например, имея кучу из 10 камней, за один ход можно получить кучу из 11 или 60 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче превышает 361. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 362 или больше камней. В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 360$ .

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.

Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причём (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.

### Пояснение.

1. а) Петя может выиграть, если  $S = 61, \dots, 360$ . При меньших значениях  $S$  за один ход нельзя получить кучу, в которой больше 360 камней. Пете достаточно увеличить количество камней в 6 раз. При  $S < 61$  получить за один ход больше 360 камней невозможно.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет  $S = 60$  камней. Тогда после первого хода Пети в куче будет 61 камень или 360 камней. В обоих случаях Ваня увеличивает количество камней в 6 раз и выигрывает в один ход.

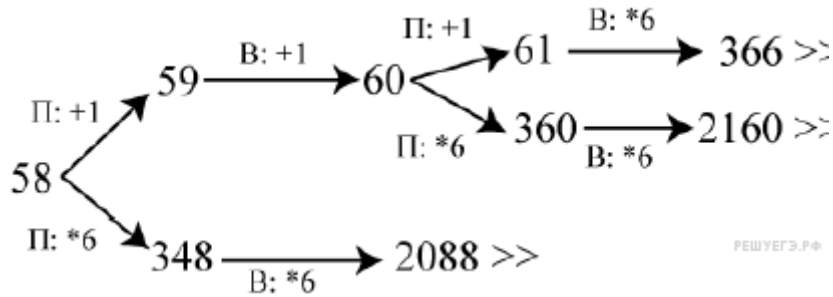
2. Возможные значения  $S$ : 10, 59. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 60 камней (при  $S = 10$  он увеличивает количество камней в 6 раз; при  $S = 59$  — добавляет 1 камень). Эта позиция разобрана в п. 1б. В ней игрок, который будет ходить (в данном случае это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.

3. Возможное значение  $S$ : 58. После первого хода Пети в куче будет 59 или 348 камней. Если в куче станет 348 камней, Ваня увеличит количество камней в 6 раз и выигрывает своим первым ходом. Ситуация, когда в куче 59 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) подчёркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

	Положения после очередных ходов			
И.п.	1-й ход Пети (разобраны все)	1-й ход Вани (только ход по стратегии)	2-й ход Пети (разобраны все)	2-й ход Вани (только ход по стратегии)

	ходы)		ходы)	
<b>58</b>	$58 + 1 = \mathbf{59}$	$59 + 1 = \mathbf{60}$	$60 + 1 = \mathbf{61}$	$61 * 6 = \mathbf{366}$
			$60 * 6 = \mathbf{360}$	$360 * 6 = \mathbf{2160}$
	$58 * 6 = \mathbf{348}$	$348 * 6 = \mathbf{2088}$		



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

**Задание 27 (С4) № 3632**

Критерии оценивания выполнения задания	Баллы
Программа работает для любых входных данных произвольного размера и находит ответ, не сохраняя входные данные в массиве, размер которого соответствует числу N (количеству запросов). Программа просматривает входные данные один раз. 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 входные данные запоминаются в массиве, размер которого соответствует числу N. Этот массив, возможно, потом сортируется. Допускается наличие от одной до трех синтаксических ошибок. Возможно, в принципиально верно организованном вводе данных есть одна ошибка (например, использование read вместо readln в Паскале или неверное считывание строки в C++). Три балла также выставляется, если в эффективной программе, удовлетворяющей критериям выставления 4 баллов, есть одна ошибка, в результате которой программа работает неверно на некоторых наборах нетипичных входных данных.	3
Программа работает в целом верно, эффективно или нет, но в реализации алгоритма содержится до двух ошибок (неверная инициализация счётчиков, допущена ошибка в принципиально верно организованной сортировке или алгоритме поиска минимальных элементов, используется знак "<" вместо "<=", "or" вместо "and" и тому подобное). Возможно, некорректно организовано считывание входных данных. Допускается наличие от одной до пяти синтаксических ошибок, описанных выше	2
Программа, возможно, неверно работает при некоторых входных данных, но по приведённому тексту решения ясно, что экзаменуемый понимает, из каких этапов должно состоять решение задачи. При использовании сортировки она может быть реализована принципиально неверно (например, вместо двух циклов используется один), или допущена принципиальная ошибка в поиске нужных элементов. Всего допускается до 4 различных ошибок в реализации алгоритма, в том числе описанных в критериях присвоения двух баллов. Допускается наличие от одной до семи синтаксических ошибок, описанных выше.	1
Задание не выполнено или выполнено неверно.	0
<i>Максимальный балл</i>	4

Вам необходимо написать программу распознавания чисел, записанных прописью. Сначала на вход программе подаётся обучающий блок, состоящий из 27 строк. Первые 9 строк содержат слова "один", "два", ..., "десять", следующие 9 строк — слова "одиннадцать", "двенадцать", ... "девятнадцать", следующие 9 строк — слова "десять", "двадцать", ..., "девяносто". Все слова записаны маленькими русскими буквами без лишних пробелов в начале и в конце строки.

Затем на вход программе подаётся значение  $N$  — количество записей, которые необходимо обработать. Следующие  $N$  строк содержат записанные словами числа. Каждое число записано по-русски, маленькими буквами, без ошибок. Если число состоит из нескольких слов, между словами находится ровно один пробел, лишних пробелов в начале и в конце строк нет.

Напишите эффективную программу, которая определит сумму тех входных чисел, которые находятся в интервале от 1 до 99.

Размер памяти, которую использует Ваша программа, не должен зависеть от длины исходного списка.

Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

*Пример входных данных (обучающий блок показан в примере с сокращениями):*

один

два

...

девяноста

5

двадцать восемь

два миллиона

четырнадцать

сто двадцать три

тысяча девятьсот восемьдесят четыре

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

42

### Пояснение.

Вариант 1.

Программа читает обучающий блок и запоминает написание чисел и их значения. Допускается построение полного "словаря" всех чисел от 1 до 99 или хранение только исходного обучающего блока. В приведённом примере на Паскале реализовано построение полного "словаря".

Затем программа читает входные строки, не запоминая их в массиве. Если построен полный "словарь", прочитанная строка ищется в этом словаре как единое целое. Числовое значение введённой строки равно сумме значений составляющих слов. Если вся строка при поиске в полном "словаре" отсутствует в обучающих данных, введённое число не попадает в интервал от 1 до 99 и не должно учитываться. Дополнительная проверка вхождения числа в заданный интервал не требуется, т. к. все числа, которые удаётся распознать с помощью приведённого обучающего блока, автоматически в него попадают, но за наличие такой дополнительной проверки в программе оценка снижается.

Паскаль
<pre> program c4; program c4; var w: array[1..99] of string;   N: integer; s: integer; line: string; i, j: integer; begin {читаем обучающий блок} for i := 1 to 9 do readln(w[i]); for i := 11 to 19 do readln(w[i]); for i := 1 to 9 do readln(w[10*i]); {строим полный словарь} for i := 2 to 9 do begin for j := 1 to 9 do begin w[10*i + j] := w[10*i] + ' ' + w[j]; end; end; {читаем и обрабатываем основной набор данных} readln(N); s:= 0 ; for i:=1 to N do begin readln(line); j:=1; while (j&lt;100) and (w[j]&lt;&gt;line) do j:=j+1; if j&lt;100 then s:=s+j; end; </pre>



```
writeln(s);
end.
```

### Вариант 2.

Программа читает обучающий блок и запоминает написание чисел и их значения. Допускается построение полного "словаря" всех чисел от 1 до 99 или хранение только исходного обучающего блока. В приведённом примере на Паскале — хранение исходного обучающего блока. Если полный "словарь" отсутствует, строка разбивается на слова (в подходящей строке их может быть не более двух). Числовое значение введённой строки равно сумме значений составляющих слов. Если какое-то слово отсутствует в обучающих данных, введённое число не попадает в интервал от 1 до 99 и не должно учитываться. Дополнительная проверка вхождения числа в заданный интервал не требуется, т. к. все числа, которые удаётся распознать с помощью приведённого обучающего блока, автоматически в него попадают, но за наличие такой дополнительной проверки в программе оценка снижается.

Паскаль

```
program c4;
const nw=27;
var
words: array[1..nw] of string;
values: array[1..nw] of integer;
{поиск слова в словаре, возврат числового значения}
function word2value (w: string) : integer;
var
i: integer;
begin
i:=1;
while (i<=nw) and (words[i]<>w) do i:=i+1;
if i<=nw then word2value:=values[i]
else word2value:=0;
end;
var
N: integer;
s: integer;
v1, v2: integer;
line: string;
i, j: integer;
begin
{читаем обучающий блок}
for i := 1 to 9 do begin
readln(words[i] );
values[i] :=i ;
end;
for i := 10 to 18 do begin
readln(words[i]);
values[i]:=i+1;
end;
for i := 19 to 27 do begin
readln(words[i]);
values[i] :=10*(i-18) ;
end;
{читаем и обрабатываем основной набор данных}
readln(N);
s:= 0 ;
for i:=1 to N do begin
readln(line);
j:=pos(line, ' ');
if j>0 then begin
v1:= word2value(copy(line,1,j-1));
v2:= word2value(copy(line,j+1,length(line)-j));
if (v1>0) and (v2>0) then s:=s+v1+v2;
end
else begin
v1:= word2value(line);
if v1>0 then s:=s+v1;
end;
end;
writeln(s);
```

end.

#### Бейсик

```

DIM w$(99)
  FOR i = 1 TO 9
    LINE INPUT w$(i)
  NEXT i
  FOR i = 11 TO 19
    LINE INPUT w$(i)
  NEXT i
  FOR i = 10 TO 90 STEP 10
    LINE INPUT w$(i)
  NEXT i
  FOR i = 20 TO 90
    FOR j = 1 TO 9
      w$(i + j) = w$(i) + " " + w$(j)
    NEXT j
  NEXT i
  INPUT N
  s = 0
  FOR i = 1 TO N
    LINE INPUT line$
    j = 1
    WHILE j < 100 AND w$(j) <> line$
      j = j + 1
    WEND
    IF j < 100 THEN s = s + j
  NEXT i
  PRINT s
END

```

#### Алгоритмический язык

```

алг с4 нач
литтаб слова[1:99]
лит трока
цел N
цел сумма
цел i, j
нц для i от 1 до 9
ввод слова[i]
кц
нц для i от 11 до 19
ввод слова[i]
кц
нц для i от 10 до 90 шаг 10
ввод слова[i]
кц
нц для i от 20 до 90 шаг 10
нц для j от 1 до 9
кц
кц
ввод N
сумма := 0
нц для i от 1 до N
ввод строка
j := 1
нц пока j < 100 и слова [j] <> строка
j := j + 1
кц
если j < 100 то сумма := сумма + j все
кц вывод
кон

```

#### Перл

```

for $v (1..9, 11..19, 20, 30, 40, 50, 60, 70, 80, 90) {
  chomp ($w = <>);
  $value{$w} = $v; }

```

```
$n = <>;
$s = 0;
while ($ n--) {
  chomp ($line = <>);
  $num = 0;
  $value{$w}
  for $w (split //,$line) {
    if ($value{$w}) {$num += $value{$w}}
    else {$num=0; last}
  }
  $s += $num;
}
print $s;
```