

Решения

↑ Задание 1 № 7185 тип 1

Сколько значащих цифр содержит двоичная запись десятичного числа 16?

Пояснение.

Запишем число 16_{10} в двоичной системе счисления: $16_{10} = 1\ 0000_2$. В двоичной записи числа 16_{10} 5 значащих цифр.

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

↑ Задание 2 № 13532 тип 2

Запишите в ответ цифры, расположив их в порядке, соответствующем буквам:

Логическая функция F задаётся выражением:

$$\neg z \wedge (\neg x \vee y).$$

На рисунке приведён фрагмент таблицы истинности функции F , содержащий все наборы аргументов, при которых функция F истинна. Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных x, y, z .

Перем. 1	Перем. 2	Перем. 3	Функция
???	???	???	F
0	0	0	1
0	0	1	1
0	1	1	1

В ответе напишите буквы x, y, z в том порядке, в котором идут соответствующие им столбцы (сначала буква, соответствующая первому столбцу, затем — буква, соответствующая второму столбцу, и т. д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

Пример. Пусть задано выражение $x \rightarrow y$, зависящее от двух переменных — x и y , и таблица истинности:

Перем. 1	Перем. 2	Функция
???	???	F
0	0	1
0	1	0
1	0	1
1	1	1

Тогда первому столбцу соответствовала бы переменная y , а второму столбцу — переменная x . В ответе следовало бы написать: yx .

Пояснение.

$$\neg z \wedge (\neg x \vee y)$$

1) Для выполнения таблицы нужно, чтобы $\neg z$ всегда был 1, значит, $z = 0$ (первый столбец).

2) Для выполнения второго условия не подходит вариант $x = 1$, а $y = 0$. Сопоставим x — второй столбец, а y — оставшийся третий.

Ответ: zxy .

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

↑ Задание 3 № 3796 тип 3

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

	A	B	C	D	E	F
A			3			
B			9		4	
C	3	9		3	8	
D			3		2	
E		4	8	2		7
F					7	

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

Пояснение.

Найдём все варианты маршрутов из A в F и выберем самый короткий.

В пункт F можно попасть только из пункта E.
 В пункт E можно попасть из пунктов C, B и D.
 В пункты B и D можно попасть только из пункта C.
 В пункт C можно попасть только из пункта A.

A-C-E-F. Длина маршрута $3 + 8 + 7 = 18$.

A-C-D-E-F. Длина маршрута $3 + 3 + 2 + 7 = 15$.

A-C-B-E-F. Длина маршрута $3 + 9 + 4 + 7 = 23$.

Видно, что кратчайший путь равен 15.

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

↑ Задание 4 № 14690 тип 4

Даны фрагменты двух таблиц из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. На основании приведённых данных определите, у скольких детей в момент рождения матерям было меньше 30 полных лет. При вычислении ответа учитывайте только информацию из приведённых фрагментов таблиц.

Таблица 1				Таблица 2	
ID	Фамилия_И. О.	Пол	Год рождения	ID_Родителя	ID_Ребенка
866	Кравец Д.К.	Ж	1942	866	911
867	Тошич Б.Ф.	М	1938	866	938
879	Гонтарь В.А.	Ж	1998	867	911
885	Крон К.Г.	М	1990	867	938
900	Кислюк Л.А.	Ж	2012	911	879
904	Петраш А.И.	М	1981	911	1041

911	Тошич А.Б.	Ж	1971	1026	900
932	Петраш П.А.	Ж	2016	938	995
938	Тошич И.Б.	М	1974	938	1017
949	Седых Г.Р.	Ж	1966	949	995
970	Кислюк А.П.	М	1968	949	1017
995	Тошич Т.И.	Ж	2002	970	879
1017	Тошич П.И.	М	2003	970	1041
1026	Мухина Р.Г.	Ж	1983	1026	932
1041	Сайко М.А.	Ж	2010	904	900
1056	Кислюк П.А.	М	1989	904	932
...

Пояснение.

Из таблицы 1 проверяем только женский пол, если его номер есть во второй таблице, в столбце родителей. Таких пар восемь: 866—911, 866—938, 911—879, 911—1041, 949—995, 949—1017, 1026—900 и 1026—932. Далее вычисляем разницу в годах рождения в этих парах и берем те, у которых получилась меньше 30.

Таких 3 пары: 866—911, 911—879 и 1026—900.

Ответ: 3.

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

↑ Задание 5 № 13481 тип 5

Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г, Д, Е, решили использовать неравномерный двоичный код, удовлетворяющий условию Фано. Для букв А, Б, В, Г использовали соответственно кодовые слова 000, 001, 10, 11. Укажите кратчайшее возможное кодовое слово для буквы Д, при котором код будет допускать однозначное декодирование. 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.

Однозначные коды не подходят по условию Фано. Кратчайшее подходящее кодовое слово — 01. Но выбирая его, не останется вариантов закодировать букву Е, значит, нужно взять минимум трехзначный код. Минимальный из них, подходящий по условию Фано — 010. Тогда букву Е можно закодировать как 011.

Ответ: 010.

Ваш ответ: *нет ответа*. Правильный ответ: 010

↑ Задание 6 № 13617 тип 6

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

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

Пример. Исходное число: 179. Произведения: $1 \cdot 7 = 7$; $7 \cdot 9 = 63$. Результат: 637.

Укажите наименьшее число, при обработке которого автомат выдаёт результат 123.

Пояснение.

Получаем произведение $x_1 \cdot x_2 = 12$, а другое произведение $x_2 \cdot x_3$ равно 3.
 $12 = 6 \cdot 2 = 3 \cdot 4$.

$3 = 1 * 3$ и никак иначе.

Значит, необходимо брать 12 как $3 * 4$ или в нашем числе будет 4 цифры, а этого не может быть. Значит, нам подходит число 134, либо число 431. Меньшее из них — 134.

Ответ: 134.

Ваш ответ: *нет ответа*. Правильный ответ: 134

↑ Задание 7 № 7252 тип 7

На рисунке приведен фрагмент электронной таблицы. Определите, чему будет равно значение, вычисленное по следующей формуле $=\text{СУММ}(B1:C4)+F2 * E4 - A3$

	A	B	C	D	E	F
1	1	3	4	8	2	0
2	4	-5	-2	1	5	5
3	5	5	5	5	5	5
4	2	3	1	4	4	2

Пояснение.

Значение $\text{СУММ}(B1:C4)$ будет равно сумме значений ячеек B1, B2, B3, B4, C1, C2, C3, C4. $\text{СУММ}(B1:C4)=14$.

Значение всей формулы $=\text{СУММ}(B1:C4)+F2 * E4 - A3$ равно $(14+5*4-5) = 29$.

Ваш ответ: *нет ответа*. Правильный ответ: 29

↑ Задание 8 № 10498 тип 8

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на пяти языках программирования.

Бейсик	Python
<pre> DIM S, N AS INTEGER S = 0 N = 0 WHILE S < 65 S = S + 6 N = N + 2 WEND PRINT N </pre>	<pre> s = 0 n = 0 while s < 65: s = s + 6 n = n + 2 print(n) </pre>
Паскаль	Алгоритмический язык
<pre> var s, n: integer; begin s := 0; n := 0; while s < 65 do s := s + 6; n := n + 2; end; writeln(n) end. </pre>	<pre> алг нач цел n, s s := 0 n := 0 нц пока s < 65 s := s + 6 n := n + 2 кц вывод n кон </pre>
Си++	

```

#include <iostream>
using namespace std;
int main()
{
    int s = 0, n = 0;
    while (s < 65)
    {
        s = s + 6;
        n = n + 2;
    }
    cout << n << endl;
    return 0;
}

```

Пояснение.

На каждой итерации цикла s увеличивается на 6. Цикл выполняется, пока s меньше 65. Значит, всего цикл сделает $(65 - 1 - 0) / 6 + 1 = 11$ итераций. Каждую итерацию n увеличивается на 2, то есть в конце $n = 0 + 2 \cdot 11 = 22$.

Ваш ответ: *нет ответа*. Правильный ответ: 22

↑ Задание 9 № 6783 тип 9

Документ объёмом 10 Мбайт можно передать с одного компьютера на другой двумя способами:

А. Сжать архиватором, передать архив по каналу связи, распаковать.

Б. Передать по каналу связи без использования архиватора.

Какой способ быстрее и на сколько, если

- средняя скорость передачи данных по каналу связи составляет 2^{23} бит в секунду,
- объём сжатого архиватором документа равен 30% от исходного,
- время, требуемое на сжатие документа, — 18 секунд, на распаковку — 2 секунды?

В ответе напишите букву А, если способ А быстрее, или Б, если быстрее способ Б. Сразу после буквы напишите, на сколько секунд один способ быстрее другого. Например, если способ Б быстрее способа А на 23 секунды, в ответе нужно написать Б23. Слов «секунд», «сек.», «с» к ответу добавлять не нужно.

Пояснение.

Общее время складывается из времени сжатия, распаковки и передачи. Время передачи t рассчитывается по формуле $t = Q / q$, где Q — объём информации, q — скорость передачи данных.

Способ А. Найдём сжатый объём: $10 \cdot 0,3 = 3$ Мбайт. Переведём Q из Мбайт в биты: 3 Мбайт = $3 \cdot 2^{20}$ байт = $3 \cdot 2^{23}$ бит. Найдём общее время: $t = 18 \text{ с} + 2 \text{ с} + 3 \cdot 2^{23} \text{ бит} / 2^{23} \text{ бит/с} = 20 + 3 \text{ с} = 23 \text{ с}$.

Способ Б. Общее время совпадает с временем передачи: $t = 10 \cdot 2^{23} \text{ бит} / 2^{23} \text{ бит/с} = 10 \text{ с}$.

Таким образом, способ Б быстрее на $23 - 10 = 13 \text{ с}$.

Ответ: Б13.

Ваш ответ: *нет ответа*. Правильный ответ: Б13

↑ Задание 10 № 5211 тип 10

Сколько есть различных символьных последовательностей длины от одного до четырёх в трёхбуквенном алфавите {А, В, С}?

Пояснение.

Если в алфавите M символов, то количество всех возможных «слов» (сообщений) длиной N равно $Q = M^N$. Так как длина слова от одного символа до четырёх, необходимо сложить количество одно-, двух-, трех- и четырехбуквенных слов.

$N=1, M=3$. Следовательно, $Q = 3^1 = 3$.

$N=2, M=3$. Следовательно, $Q = 3^2 = 9$.

$N=3, M=3$. Следовательно, $Q = 3^3 = 27$.

$N=4, M=3$. Следовательно, $Q = 3^4 = 81$.

Следовательно, ответ 120.

Ваш ответ: *нет ответа*. Правильный ответ: 120

↑ Задание 11 № 15948 тип 11

Ниже на пяти языках программирования записан рекурсивный алгоритм F.

Бейсик	Python
<pre>SUB F(n) IF n > 0 THEN PRINT N F(n \ 2) F(n - 4) END IF END SUB</pre>	<pre>def F(n): if n > 0: print(n) F(n // 2) F(n - 4) else: return n</pre>
Паскаль	Алгоритмический язык
<pre>procedure F(n: integer); begin if n > 0 then begin write(n); F(n div 2); F(n - 4); end end;</pre>	<pre>алг F(цел n) нач если n > 0 то вывод n F(div(n,2)) F(n - 4) все кон</pre>
C++	
<pre>void F (int n) { if (n > 0) { std::cout << n; F (n / 2); F (n - 4); } }</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) выведет на экран число 9, после этого процедура F(9) вызовет процедуру F(4), которая выведет на экран число 4 и вызовет процедуру F(2), которая выведет на экран число 2 и вызовет процедуру F(1), которая выведет на экран число 1. После этого управление вернётся к процедуре F(9), которая вызовет процедуру F(5), которая выведет на экран число 5 и вызовет процедуру F(2), после чего на экран будет выведено число 2 и будет вызвана процедура F(1), которая выведет на экран число 1.

После этого управление вернётся к процедуре F(5), которая следующим шагом своего алгоритма вызовет процедуру F(1), которая выведет на экран число 1.

Ответ: 94215211.

Ваш ответ: *нет ответа*. Правильный ответ: 94215211

↑ Задание 12 № 3510 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: 217.9.142.131

Маска: 255.255.192.0

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

A	B	C	D	E	F	G	H
0	9	16	64	128	142	192	217

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 142.

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

$$142_{10} = 10001110_2$$

Результатом конъюнкции является число $10000000_2 = 128$.

4. Сопоставим варианты ответа получившимся числам: 217, 9, 128, 0.

Ответ: HBEA.

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

↑ Задание 13 № 217 тип 13

В скачках участвуют 20 лошадей. Специальное устройство регистрирует прохождение каждой лошадию финиша, записывая ее номер с использованием минимально возможного количества бит, одинакового для каждой лошади. Каков информационный объем сообщения, записанного устройством, если до финиша добрались только 15 из 20 участвовавших в скачках лошадей? (Ответ дайте в битах.)

Пояснение.

Известно, что с помощью N бит можно закодировать 2^N различных чисел. Поскольку $2^4 < 20 < 2^5$, то для записи каждого из 20 номеров необходимо 5 бит памяти. Поскольку до финиша добрались только 15, то информационный объем сообщения составит $15 \cdot 5 = 75$ бит.

Ваш ответ: *нет ответа*. Правильный ответ: 75

↑ Задание 14 № 5900 тип 14

Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости, включает в себя 4 команды-приказа и 4 команды проверки условия.

Команды-приказы:

вверх	вниз	влево	вправо
--------------	-------------	--------------	---------------

При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

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

Другие 4 команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

сверху свободно	снизу свободно	слева свободно	справа свободно
------------------------	-----------------------	-----------------------	------------------------

Цикл

ПОКА *условие*

последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ *условие*

ТО *команда1*

ИНАЧЕ *команда2*

КОНЕЦ ЕСЛИ

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

Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка F1)?

НАЧАЛО

ПОКА **справа свободно** ИЛИ **сверху свободно**

ЕСЛИ **справа свободно**

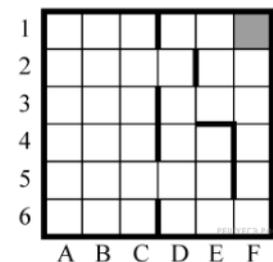
ТО **вправо**

ИНАЧЕ **вверх**

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

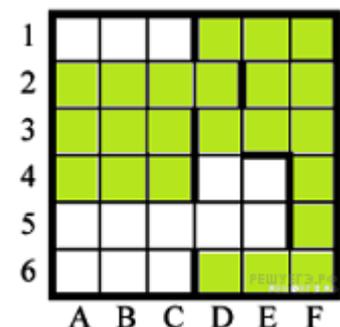


Пояснение.

При данной программе РОБОТ поступает следующим образом: сперва РОБОТ проверяет, свободна ли клетка справа или сверху от него. Если это так, то РОБОТ переходит к первому действию внутри цикла. В этом цикле если у правой стороны клетки, в которой находится РОБОТ, нет стены, он двигается вправо, в противном случае он перемещается вверх. После этого возвращается к началу внешнего цикла.

Проанализировав эту программу, приходим к выводу, что РОБОТ не может разбиться.

Проверив все клетки по выведенному нами правилу движения РОБОТА, выясняем, что число клеток, удовлетворяющих условию задачи равно 23: вся вторая и третья строчки, D1-F1, A4-C4, D6-F6, F2-F5.

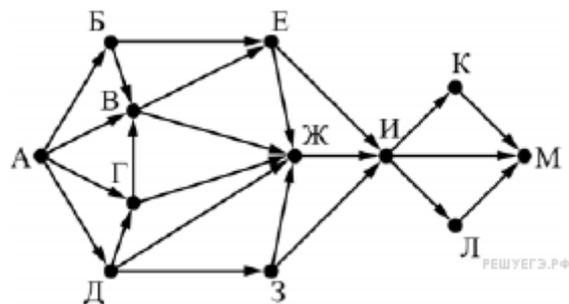


Ваш ответ: *нет ответа*. Правильный ответ: 23

↑ Задание 15 № 11351 тип 15

На рисунке представлена схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К, Л, М. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.

Сколько существует различных путей из города А в город М, проходящих через город В?

**Пояснение.**

Количество путей до города X = количество путей добраться в любой из тех городов, из которых есть дорога в X .

При этом если путь должен не проходить через какой-то город, нужно просто не учитывать этот город при подсчёте сумм. А если город наоборот обязательно должен лежать на пути, тогда для городов, в которые из нужного города идут дороги, в суммах нужно брать только этот город.

С помощью этого наблюдения посчитаем последовательно количество путей до каждого из городов:

$$A = 1$$

$$B = A = 1$$

$$D = A = 1$$

$$G = A + D = 1 + 1 = 2$$

$$V = A + B + G = 4$$

$$E = V = 4$$

$$Z = 0 \text{ (поскольку в } Z \text{ не ведёт ни одна дорога из } V)$$

$$I = E + Z = 4 + 0 = 4$$

$$K = L = I = 4$$

$$M = K + I + L = 12$$

Примечание. Необходимо найти количество различных путей из города A в город M , проходящих через город B .

Ответ: 36.

Ваш ответ: *нет ответа*. Правильный ответ: 36

↑ Задание 16 № 5058 тип 16

В некоторой системе счисления записи десятичных чисел 66 и 40 заканчиваются на 1. Определите основание системы счисления.

Пояснение.

Составим уравнение для перевода числа 66_{10} и 40_{10} в n -ую систему счисления ($n > 2$).

$$66_{10} = \dots + a \cdot n^2 + b \cdot n^1 + 1 \cdot n^0$$

$$40_{10} = \dots + c \cdot n^2 + d \cdot n^1 + 1 \cdot n^0$$

$$65_{10} = \dots + a \cdot n^2 + b \cdot n^1$$

$$39_{10} = \dots + c \cdot n^2 + d \cdot n^1$$

Так как n — целое, 65 и 39 должно делиться нацело на n . Общий делитель этих двух чисел единственен — 13.

Ответ: 13.

Ваш ответ: *нет ответа*. Правильный ответ: 13

↑ Задание 17 № 6429 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)
Евклид & Аристотель	240
Евклид & (Аристотель Платон)	450
Евклид & Аристотель & Платон	90

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

Евклид & Платон.

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.

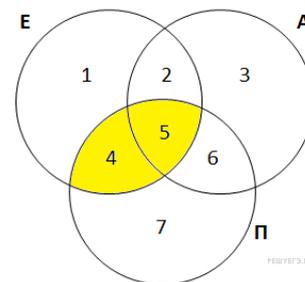
Количество запросов в данной области будем обозначать N_i . Наша цель — $N_4 + N_5$. Тогда из таблицы находим, что:

$$\begin{aligned} N_2 + N_5 &= 240, \\ N_5 &= 90, \\ N_2 + N_4 + N_5 &= 450. \end{aligned}$$

Из первого и второго равенств находим: $N_2 = 150$, из последнего равенства: $N_4 + N_5 = 300$.

Ответ: 300.

Ваш ответ: *нет ответа*. Правильный ответ: 300



↑ Задание 18 № 9653 тип 18

На числовой прямой даны два отрезка: $P = [10, 29]$ и $Q = [13, 18]$.

Укажите наибольшую возможную длину отрезка A , для которого выражение

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

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

Пояснение.

Введем обозначения:

$$(x \in A) \equiv A; (x \in P) \equiv P; (x \in Q) \equiv Q.$$

Применив преобразование импликации, получаем:

$$\neg A \vee P \vee Q$$

Логическое ИЛИ истинно, если истинно хотя бы одно утверждение. Выражение $P \vee Q$ истинно на отрезке $[10, 29]$. Значит, $\neg A$ должно быть истинно вне этого отрезка, следовательно, A должно быть истинно на отрезке $[10, 29]$. Его длина 19.

Ответ: 19.

Ваш ответ: *нет ответа*. Правильный ответ: 19

↑ **Задание 19 № 15804 тип 19**

Представленный ниже фрагмент программы обрабатывает элементы одномерного целочисленного массива A с индексами от 1 до 10. Перед началом выполнения данного фрагмента эти элементы массива имели значения 2, 3, 4, 4, 10, 4, 5, 6, 12, 9 (т.е. $A[1] = 2$, $A[2] = 3$, ..., $A[10] = 9$). Определите значение переменной s после выполнения фрагмента этой программы (записанного ниже на разных языках программирования).

Бейсик	Python
<pre>n = 10 s = 0 FOR i = 2 TO n IF A(i-1) < A(i) THEN T = A(i-1) A(i-1) = A(i) A(i) = T + 1 s = s + 1 END IF NEXT i</pre>	<pre>n = 10 s = 0 for i in range(2,n+1): if A[i-1] < A[i]: t = A[i-1] A[i-1] = A[i] A[i] = t + 1 s = s + 1</pre>
Паскаль	Алгоритмический язык
<pre>n := 10; s := 0; for i:=2 to n do begin if A[i-1] < A[i] then begin t := A[i-1]; A[i-1] = A[i]; A[i] := t + 1; s := s + 1 end end;</pre>	<pre>n := 10 s := 0 нц для i от 2 до n если A[i-1] < A[i] то t := A[i-1] A[i-1] = A[i] A[i] := t + 1 s := s + 1 все кц</pre>
C++	
<pre>n = 10; s = 0; for (i = 2; i <= n; ++i) { if (A[i-1] < A[i]) { t = A[i-1]; A[i-1] = A[i]; A[i] = t + 1; s = s + 1; } }</pre>	

Пояснение.

Данный алгоритм делает предыдущий элемент равным текущему, а текущему элементу присваивается значение предыдущего элемента, к которому прибавляется единица, после чего к переменной s прибавляется единица, если предыдущее значение элемента меньше текущего.

Изначальный порядок значений: 2, 3, 4, 4, 10, 4, 5, 6, 12, 9.

Первое изменение элементов: 3, **3**, 4, 4, 10, 4, 5, 6, 12, 9.

Второе изменение элементов: 3, 4, **4**, 4, 10, 4, 5, 6, 12, 9.

Третье изменение элементов: 3, 4, 4, 10, **5**, 4, 5, 6, 12, 9.

Четвёртое изменение элементов: 3, 4, 4, 10, 5, 5, **5**, 6, 12, 9.

Пятое изменение элементов: 3, 4, 4, 10, 5, 5, 6, **6**, 12, 9.

Шестое изменение элементов: 3, 4, 4, 10, 5, 5, 6, 12, **7**, 9.

Последнее изменение элементов: 3, 4, 4, 10, 5, 5, 6, 12, 9, **8**.

Всего изменение элементов массива произошло семь раз. Следовательно, переменная $s = 7$.

Ответ: 7.

Ваш ответ: *нет ответа*. Правильный ответ: 7

↑ Задание 20 № 5748 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 12, а потом 5.

Бейсик	Паскаль
<pre> DIM X, A, B, C AS INTEGER INPUT X A = 0: B = 10 WHILE X > 0 C = X MOD 10 A = A + C IF C < 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>0 do begin c := x mod 10; a := a+c; if c < b then b := c; x := x div 10; end; writeln(a); write(b); end. </pre>
Си++	Алгоритмический
<pre> #include <iostream> using namespace std; int main() { int x, a, b, c; cin >> x; a = 0; b = 10; while (x>0) { c = x%10; a = a+c; if (c<b) b = c; x = x /10; } cout << a << endl << b endl; } </pre>	<pre> алг нач цел x, a, b, c ввод X a := 0; b := 10 нц пока x>0 c := mod(x,10) a := a+c если c<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 > 0: c = x % 10 a += c if c < 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 12, необходимо, чтобы сумма цифр числа x была равна 12. Поскольку вторым должно быть напечатано число 5, необходимо, чтобы первой цифрой числа x была цифра 5. Наименьшее такое число — 57.

Ответ: 57.

Ваш ответ: *нет ответа*. Правильный ответ: 57

↑ Задание 21 № 3330 тип 21

Определите, какое число будет напечатано в результате выполнения следующего алгоритма:

Бэйсик	Паскаль
<pre>DIM A, B, T, M, R AS INTEGER A = -20: B = 20 M = A: R = F(A) FOR T = A TO B IF F(T) < R THEN M = T R = F(T) END IF NEXT T PRINT R FUNCTION F(x) F := 4*(x-5)*(x+3) END FUNCTION</pre>	<pre>var a,b,t,M,R :integer; Function F(x: integer):integer; begin F := 4*(x-5)*(x+3); end; BEGIN a := -20; b := 20; M := a; R := F(a); for t := a to b do begin if (F(t)< R) then begin M := t; R := F(t); end; end; write(R); END.</pre>
Си++	Алгоритмический
<pre>#include <iostream> using namespace std; int F(int x) { return 4*(x-5)*(x+3) } int main() { int a, b, t, M, R; a = -20; b = 20; M = a; R = F(a); for (t=a; t<=b; t++){ if (F(t) < R) { M = t; R = F(t); } } cout << R << endl; }</pre>	<pre>алг нач цел a, b, t, R, M a := -20; b := 20 M := a; R := F(a) нц для t от a до b если F(t) < R то M := t; R := F(t) все кц вывод R кон алг цел F(цел x) нач знач := 4*(x-5)*(x+3) кон</pre>

```

Python

def f(x):
    return 4*(x-5)*(x+3)
a = -20
b = 20
M = a
R = f(a)
for t in range(a, b+1):
    if (f(t) < R):
        M = t
        R = f(t);
print(R)

```

Пояснение.

1. Алгоритм предназначен для поиска наименьшего значения функции $F(t)$ на отрезке от a до b .

2. $F(x) = 4(x-5)(x+3)$ Квадратный трехчлен $F(t)$ с положительным старшим коэффициентом пересекает ось абсцисс в точках 5 и -3 и, следовательно, наименьшее значение достигается в вершине 1 и равно $F(1) = -64$.

Ваш ответ: *нет ответа*. Правильный ответ: -64

↑ Задание 22 № 15144 тип 22

Исполнитель Фибо преобразует число на экране.

У исполнителя есть две команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 2.

Программа для исполнителя Фибо — это последовательность команд.

Сколько существует программ, которые преобразуют исходное число 2 в число 18 и при этом траектория вычислений содержит число 9 и не содержит числа 14?

Траектория вычислений — это последовательность результатов выполнения всех команд программы. Например, для программы 212 при исходном числе 7 траектория будет состоять из чисел 9, 10, 12.

Пояснение.

Искомое количество программ равно произведению количества программ, получающих из числа 2 число 9, на количество программ, получающих из числа 9 число 13 и на количество программ, получающих из числа 15 число 18, поскольку траектория вычислений не должна содержать числа 14.

Пусть $R(n)$ — количество программ, которые число 2 преобразуют в число n , $P(n)$ — количество программ, которые число 9 преобразуют в число n , а $F(n)$ — количество программ, которые преобразуют число 15 в число n .

Для всех $n > 4$ верны следующие соотношения:

1. $R(n) = R(n-1) + R(n-2)$, так как существует два способа получения n — прибавлением единицы или прибавлением двойки. Аналогично $P(n) = P(n-1) + P(n-2)$ и $F(n) = F(n-1) + F(n-2)$.

Последовательно вычислим значения $R(n)$:

$$\begin{aligned}
 R(2) &= 1. \\
 R(3) &= 1. \\
 R(4) &= R(2) + R(3) = 2. \\
 R(5) &= R(3) + R(4) = 3. \\
 R(6) &= R(5) + R(4) = 5. \\
 R(7) &= R(6) + R(5) = 8. \\
 R(8) &= R(7) + R(6) = 13. \\
 R(9) &= R(8) + R(7) = 21.
 \end{aligned}$$

Теперь вычислим значения $P(n)$:

$$\begin{aligned}
 P(9) &= 1. \\
 P(10) &= 1. \\
 P(11) &= P(9) + P(10) = 2. \\
 P(12) &= P(10) + P(11) = 3.
 \end{aligned}$$

$$P(13) = P(11) + P(12) = 5.$$

Теперь вычислим значения $F(n)$:

$$\begin{aligned} F(15) &= 1. \\ F(16) &= 1. \\ F(17) &= F(15) + F(16) = 2. \\ F(18) &= F(16) + F(17) = 3. \end{aligned}$$

Таким образом, количество программ, удовлетворяющих условию задачи равно $21 \cdot 5 \cdot 3 = 315$.

Ответ: 315.

Ваш ответ: *нет ответа*. Правильный ответ: 315

↑ Задание 23 № 7468 тип 23

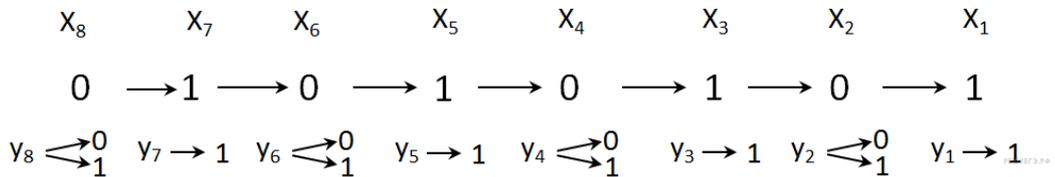
Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, которые удовлетворяют всем перечисленным ниже условиям?

$$\begin{aligned} (x_1 \vee x_2) \wedge ((x_1 \wedge x_2) \rightarrow x_3) \wedge (\neg x_1 \vee y_1) &= 1 \\ (x_2 \vee x_3) \wedge ((x_2 \wedge x_3) \rightarrow x_4) \wedge (\neg x_2 \vee y_2) &= 1 \\ \dots \\ (x_6 \vee x_7) \wedge ((x_6 \wedge x_7) \rightarrow x_8) \wedge (\neg x_6 \vee y_6) &= 1 \\ (x_7 \vee x_8) \wedge (\neg x_7 \vee y_7) &= 1 \\ (\neg x_8 \vee y_8) &= 1 \end{aligned}$$

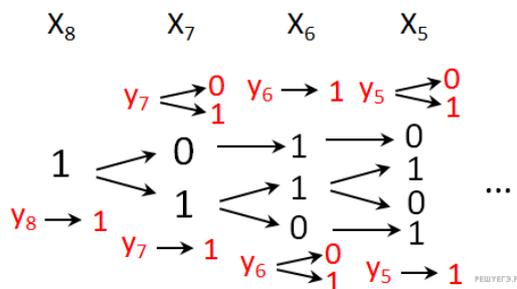
В ответе не нужно перечислять все различные наборы значений переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

Пояснение.

Из последнего уравнения находим, что возможны три варианта значений x_8 и y_8 : 01, 00, 11. Построим дерево вариантов для первой и второй пар значений.



Таким образом, имеем 16 наборов переменных. Дерево вариантов для пары значений 11:



Получаем 45 вариантов. Таким образом, система будет иметь $45 + 16 = 61$ различных наборов решений.

Ответ: 61.

Ваш ответ: *нет ответа*. Правильный ответ: 61

Проверка части с развернутым ответом

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

Задание 24 (С1) № 14710

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

Факториалом натурального числа n (обозначается $n!$) называется произведение всех натуральных чисел от 1 до n . Например, $4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24$.

Дано целое положительное число A . Необходимо найти минимальное натуральное K , для которого $K! \geq A$.

Для решения этой задачи ученик написал программу, но, к сожалению, его программа неправильная.

Ниже эта программа для Вашего удобства приведена на пяти языках программирования.

Бейсик	Python
<pre> DIM A, K, F AS INTEGER INPUT A K = 0 F = 1 WHILE F <= A K = K + 1 F = F * K WEND PRINT K END </pre>	<pre> a = int(input()) k = 0 f = 1 while f <= a: k += 1 f *= k print(k) </pre>
Паскаль	Алгоритмический язык
<pre> var a, k, f: integer; begin read(a); k := 0; f := 1; while f <= a do begin k := k + 1; f := f * k; end; writeln(k) end. </pre>	<pre> алг нач цел a, k, f ввод a k := 0 f := 1 нц пока f <= a k := k + 1 f := f * k кц вывод k кон </pre>
Си++	
<pre> #include using namespace std; int main(){ int a, k, f; cin >> a; k = 0; f = 1; while (f <= a) { ++k; f *= k; } cout << k; } </pre>	

```

    return 0;
}

```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе $A = 6$.
2. Назовите **минимальное A , большее 10**, при котором программа выведет **неверный** ответ.
3. Найдите в программе все ошибки (их может быть одна или несколько).

Для каждой ошибки выпишите строку, в которой она допущена, и приведите эту же строку в исправленном виде.

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

Обратите внимание: Вам нужно исправить приведённую программу, а не написать свою. Вы можете только заменять ошибочные строки, но не можете удалять строки или добавлять новые. Заменять следует только ошибочные строки: за исправления, внесённые в строки, не содержащие ошибок, баллы будут снижаться.

Пояснение.

1. При вводе $A = 6$ программа выведет число 4.

2. Минимальное $A > 10$, при котором программа выдаёт неверный ответ, равно 24. Программа выдаёт **неверные** ответы, если введённое число оказывается точным факториалом. Минимальный точный факториал, больший 10, равен 24.

3. Программа содержит две ошибки.

1) Неверная инициализация. Начальное значение переменной K должно быть равно 1. Это значение не влияет на работу программы при $A > 1$, но при $A = 1$ оно оказывается существенным. Если исправить вторую ошибку, сохранив нулевое начальное значение K , то программа выдаст ответ 0 вместо правильного 1.

- 2) Неверное условие цикла.

Пример исправления для языка Паскаль:

Первая ошибка:

```
k := 0;
```

Исправленная строка:

```
k := 1;
```

Вторая ошибка:

```
while f <= a do begin
```

Исправленная строка:

```
while f < a do begin
```

В программах на других языках ошибочные строки и их исправления аналогичны.

Незначительной опiskой, не влияющей на оценку, следует считать отсутствие служебных слов и знаков после содержательной части исправления

Задание 25 (С2) № 5256

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 языках, использованных в задании. 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
Предложено в целом верное решение, содержащее не более одной ошибки из числа следующих: <ol style="list-style-type: none"> 1) Не инициализируется или неверно инициализируется текущее значение минимума. В частности, нельзя инициализировать это значение суммой двух первых элементов массива. 2) Неверно проверяется чётность элементов или совпадение чётности. 3) 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). 4) Вместо значения элемента проверяется его индекс. 5) Отсутствует вывод ответа. 6) Используется переменная, не объявленная в разделе описания переменных. 7) Не указано или неверно указано условие завершения цикла, например, используется цикл от 1 до N, и при обращении к элементу $a[i+1]$ происходит выход за границы массива. 8) Индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно 	1
Не выполнены условия, позволяющие поставить 1 или 2 балла	0
<i>Максимальный балл</i>	2

Дан массив, содержащий 70 неотрицательных целых чисел, не превышающих 1000. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести наименьшую сумму двух соседних элементов массива, имеющих одинаковую чётность. Гарантируется, что в массиве есть соседние элементы, имеющие одинаковую чётность. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

Паскаль	Бейсик

<pre>const N=70; var a: array [1..N] of integer; i, j, x, y: integer; begin for i:=1 to N do readln(a[i]); ... end.</pre>	<pre>N=70 DIM A(N) AS INTEGER DIM I, J, X, Y AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre>
Си++	Алгоритмический язык
<pre>#include <iostream> using namespace std; #define N 70 int main() { int a[N]; int i, j, x, y; for (i=0; i<N; i++) cin >> a[i];</pre>	<pre>алг нач цел N=70 целтаб a[1:N] цел i, j, x, y нц для i от 1 до N ввод a[i] кц ... кон</pre>
Python	
<pre># допускается также # использовать три # целочисленные переменные j, x, y a = [] n = 70 for i in range(0, n): a.append(int(input())) ...</pre>	

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

Пояснение.

Пример программы на Паскале:

```
y:=2001;
for i:=1 to N-1
do begin x:=a[i]+a [i + 1];
if (a[i] mod 2 = a[i+1] mod 2) and (x < y)
then y:=x;
end;
writeln(y);
```

Допускается использование других приёмов программирования, приводящих к верному результату.

В частности, разрешается использовать следующие способы:

1. Отказ от использования отдельной переменной для суммы двух элементов. В этом случае сумма может вычисляться повторно:

```
if (a[i] mod 2 = a[i+1] mod 2) and (a[i]+a[i+1]
```

2. Проверка совпадения чётности полным перебором вариантов:

```
if ( (a [i] mod 2 = 0) and (a[i + 1] mod 2 = 0) or (a[i] mod 2=1) and (a[i+1] mod 2=1)) and (a[i]+a[i+1]
```

3. Проверка совпадения чётности по чётности суммы:

```
if (x mod 2=0) and (x<y)
```

4. Использование вложенных условных операторов вместо составного условия:

```
if (a[i] mod 2 = a[i+1] mod 2) then if x
```

Допускаются и другие вариации при условии, что они соответствуют указанным в условии ограничениям и приводят к правильному ответу.

Задание 26 (С3) № 4736

Критерии оценивания выполнения задания	Баллы
Выполнены второе и третье задания. Первое задание выполнено полностью или частично. Здесь и далее допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному	3

ответу	
Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий. 1. Задание 3 выполнено полностью. 2. Первое и второе задания выполнены полностью. 3. Первое задание выполнено полностью или частично; для заданий 2 и 3 указаны правильные значения S	2
Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий. 1. Первое задание выполнено полностью. 2. Во втором задании правильно указано одно из двух возможных значений S и для этого значения указана и обоснована выигрышная стратегия Паши. 3. Первое задание выполнено частично и для одного из остальных заданий правильно указано значение S. 4. Для второго и третьего задания правильно указаны значения S	1
Не выполнено ни одно из условий, позволяющих поставить 3, 2 или 1 балл.	0
Максимальный балл	3

Два игрока, Петя и Ваня, играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых 4, а во второй - 3 камня. У каждого игрока неограниченно много камней. Игроки ходят по очереди, первый ход делает Петя. Ход состоит в том, что игрок или утраивает число камней в какой-то куче, или добавляет 1 камень в какую-то кучу. Игра завершается в тот момент, когда общее количество камней в двух кучах становится не менее 20. Если в момент завершения игры общее число камней в двух кучах не менее 35, то выиграл Ваня, в противном случае - Петя. Кто выигрывает при безошибочной игре обоих игроков? Укажите, стратегию выигрывающего игрока - какой ход он должен сделать в каждой из позиций, которые могут ему встретиться при правильной игре. Докажите, что описанная стратегия - выигрышная.

Пояснение.

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

Выигрывает Петя. Для того, чтобы выиграть, он должен утроить количество камней во второй куче.

Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны пары чисел, разделённые запятой. Эти числа соответствуют количеству камней на каждом этапе игры в первой и второй кучах соответственно.

1 ход	2 ход	3 ход	
Петя (выигрышный ход)	Ваня (все варианты)	Петя (выигрышный ход)	
4,9	12,9	Петя выиграл	
	4,27	Петя выиграл	
	5,9	15,9	Петя выиграл
		5,27	
	4,10	12,10	Петя выиграл
		4,30	

Таблица содержит все возможные варианты ходов второго игрока. Из неё видно, что при любом ходе второго игрока у первого имеется ход, приводящий к победе. Причём у первого игрока есть два варианта выигрышного хода. Описание любого из них является правильным решением.

Задание 27 (С4) № 6938

Критерии оценивания выполнения задания	Баллы
Программа правильно работает для любых входных данных и является эффективной как по времени, так и по памяти. 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
Не выполнены условия, позволяющие поставить 4 балла. Программа работает верно и является эффективной по времени, но размер используемой памяти зависит от количества исходных данных. Например, входные данные (все значения D и K) запоминаются в массиве или другой структуре данных, размер которой зависит от N. Допускается одна из следующих ошибок: 1) Вместо остатка от деления K на D вычисляется остаток от деления D на K. 2) Отсутствует предварительная инициализация счётчиков (для языков, в которых нет гарантированного начального обнуления). 3) Неверно разрешается конфликт равенства максимумов (выбирается меньшее значение индекса вместо большего). 4) Неверно выводится ответ (или не выводится никакого ответа), если все остатки оказались равны нулю. Допускается наличие от одной до трёх синтаксических ошибок, описанных в критериях на 4 балла.	3
Не выполнены условия, позволяющие поставить 3 или 4 балла. Программа работает в целом верно, эффективно или нет. Возможны переборные решения, при которых все исходные данные хранятся в массиве (или в двух массивах), этот массив многократно просматривается, при каждом просмотре подсчитывается количество пар с определённым остатком. В реализации алгоритма допущено более 1 ошибки из числа перечисленных в критериях на 3 балла или допущены другие ошибки, приводящие к неверной работе программы в отдельных случаях. Допускается наличие от одной до пяти синтаксических ошибок, описанных в критериях на 4 балла.	2

Не выполнены условия, позволяющие поставить 2, 3 или 4 балла. Программа работает в отдельных частных случаях. Один балл также ставится, если программа написана неверно, но из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи.	1
Не выполнены условия, позволяющие поставить 1, 2, 3 или 4 балла.	0
Максимальный балл	4

На плоскости дан набор точек с целочисленными координатами. Необходимо найти четырёхугольник наибольшей площади с вершинами в этих точках, две вершины которого лежат на оси Oy , а две оставшиеся — по разные стороны от оси Ox .

Напишите эффективную, в том числе по памяти, программу, которая будет решать эту задачу. Перед текстом программы кратко опишите алгоритм решения задачи и укажите используемый язык программирования и его версию.
Описание входных данных.

В первой строке вводится одно целое положительное число — количество точек N . Каждая из следующих N строк содержит два целых числа: сначала координата x , затем координата y очередной точки.

Описание выходных данных.

Программа должна вывести одно число — максимальную площадь четырёхугольника, удовлетворяющего условиям задачи. Если такого четырёхугольника не существует, программа должна вывести ноль.

Пример входных данных:

```
6
0 0
2 0
0 2
3 -3
-5 -5
6 6
```

Пример выходных данных для приведённого выше примера входных данных:

```
11
```

Пояснение.

Искомый четырёхугольник состоит из двух треугольников с общим основанием, лежащим на оси Ox , при этом один треугольник лежит выше этой оси, другой — ниже. Площадь четырёхугольника будет максимальной, если вершины на оси Ox будут расположены как можно дальше друг от друга, а вершины, не лежащие на этой оси, — как можно дальше от неё. Программа читает исходные данные, запоминая все точки в массиве. Для каждой точки проверяется её принадлежность оси Ox (условие $x = 0$). Среди точек, лежащих на оси, необходимо найти наиболее далеко отстоящие друг от друга — они дадут наибольшее возможное общее основание двух треугольников. Это будут точки с наименьшим и наибольшим значением координаты y . Среди точек, не лежащих на оси Ox , надо найти две точки, расположенные по разные стороны от оси и как можно дальше от неё, — они дадут наибольшие возможные значения высот треугольников. Это будут точки с наибольшим положительным и наименьшим отрицательным значением координаты x . Таким образом, задача сводится к нахождению максимального и минимального y среди точек, у которых $x = 0$, максимального и минимального x среди остальных точек и нахождению площади четырёхугольника на основе этих данных. Перед выводом результата необходимо убедиться в существовании искомого четырёхугольника.

Пример правильной и эффективной программы на языке Паскаль

```
program c4;
var
n: integer;
x, y: integer;
ymin, ymax: integer;
ysearch: boolean;
xmin, xmax: integer;
i: integer;
s: real;
begin
ysearch := true;
ymin:=0; ymax := 0;
xmin := 0; xmax := 0;
readln(n);
for i:=1 to n do begin
readln(x,y);
if x=0 then begin
if ysearch or (y < ymin) then ymin:=y;
if ysearch or (y > ymax) then ymax:=y;
ysearch:=false;
end
else if x < xmin then xmin:=x
else if x > xmax then xmax:=x
end;
if (ymax>ymin) and (xmin < 0) and (xmax>0)
then s := (xmax-xmin)*(ymax-ymin)/2
else s := 0;
writeln(s);
end.
```

Пример правильной и эффективной программы на языке Бейсик

```
DIM n AS INTEGER
```

```

DIM x, y AS INTEGER
DIM ymin, ymax AS INTEGER
DIM ysearch AS INTEGER
DIM xmin, xmax AS INTEGER
DIM i AS INTEGER
DIM s AS DOUBLE
ysearch = 1
ymin = 0: ymax = 0
xmin = 0: xmax = 0
INPUT n
FOR i = 1 TO n
INPUT x, y
IF x = 0 THEN
IF ysearch = 1 OR y < ymin THEN ymin = y
IF ysearch = 1 OR y > ymax THEN ymax = y
ysearch = 0
ELSEIF x < xmin THEN xmin = x
ELSEIF x > xmax THEN xmax = x
END IF
NEXT i
IF ymax > ymin AND xmin < 0 AND xmax > 0 THEN
s = (xmax - xmin) * (ymax - ymin) / 2
ELSE
s = 0
END IF
PRINT s

```

Пример правильной, но неэффективной программы на языке Паскаль.

```

var
coord: array[1..10000, 1..2] of integer; {исходные данные}
x, y: integer; {координаты очередной точки}
xminpos, xmaxpos, yminpos, ymaxpos: integer; {координаты точек четырёхугольника с наибольшей площадью}
s: real; {площадь четырёхугольника}
N: integer; {количество точек}
i, j: integer;
begin
readln(N);
xminpos := MaxInt; xmaxpos := -(MaxInt-1);
yminpos := MaxInt; ymaxpos := -(MaxInt-1);
for i := 1 to N do begin
read(x, y);
coord[i, 1] := x; coord[i, 2] := y;
end;
for i := 1 to N do begin
if (coord[i, 1] = 0) and (coord[i, 2] > ymaxpos) then ymaxpos := coord[i, 2];
if (coord[i, 1] <> 0) and (coord[i, 1] > xmaxpos) then xmaxpos := coord[i, 1];
end;
if (yminpos = ymaxpos) or (xminpos = xmaxpos) or (yminpos = MaxInt) then
s := 0
else s := (ymaxpos - yminpos)*(xmaxpos-xminpos)/2;
writeln(s);
end.

```