

Решения

↑ Задание 1 № 421 тип 1

Вычислите сумму чисел x и y , при $X = D6_{16}$, $Y = 36_8$. Результат представьте в двоичной системе счисления.

Пояснение.

$$\begin{aligned} D6_{16} &= 214_{10} \\ 36_8 &= 30_{10} \\ x + y &= 214 + 30 = 244 \\ 244_{10} &= 11110100_2 \end{aligned}$$

Ваш ответ: *нет ответа*. Правильный ответ: 11110100

↑ Задание 2 № 13398 тип 2

Логическая функция F задаётся выражением $(x \rightarrow y) \wedge (y \rightarrow z)$. На рисунке приведён фрагмент таблицы истинности функции F . Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных x, y, z .

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

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

Пояснение.

Раскроем две импликации:

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

Поскольку все значения F в таблице равны 1, то можно предположить, что x соответствует 3 столбцу, т.к. тогда $\neg x$ будет всегда равен 1. По той же логике z всегда выгодно брать 1 и сопоставить ему 1 столбец.

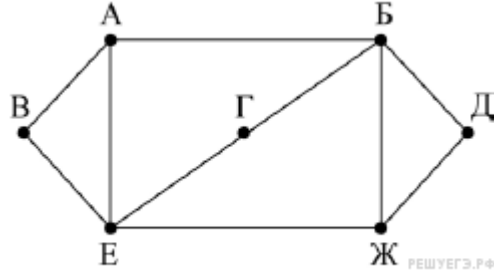
Ответ: zyx .

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

↑ **Задание 3 № 16879 тип 3**

На рисунке схема дорог изображена в виде графа, в таблице содержатся сведения о длине этих дорог в километрах.

	П1	П2	П3	П4	П5	П6	П7
П1					10		8
П2			7			6	12
П3		7				4	
П4						6	7
П5	10					15	14
П6		6	4	6	15		
П7	8	12		7	14		



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 кратчайшего пути из пункта В в пункт Д, если передвигаться можно только по указанным дорогам. В ответе запишите целое число — длину пути в километрах.

Пояснение.

Заметим, что Г — единственная вершина степени 2, которая связана с двумя вершинами степени 4. Следовательно, Г соответствует П4. Далее рассмотрим два варианта:

1. Е соответствует П6, а Б соответствует П7. Значит, В соответствует П3, а А соответствует П2. Тогда Д соответствует П1, а Ж соответствует П5. Тогда кратчайшее расстояние из пункта В в пункт Д равняется 25.

2. Е соответствует П7, а Б соответствует П6. Значит, В соответствует П1, а А соответствует П5. Тогда Д соответствует П3, а Ж соответствует П2. Тогда кратчайшее расстояние из пункта В в пункт Д равняется 25.

Ответ: 25.

Ваш ответ: *нет ответа*. Правильный ответ: 25

↑ **Задание 4 № 6564 тип 4**

Ниже представлены две таблицы из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведённых данных ID племянницы Степаненко З. А.

Пояснение: племянницей считается дочь брата или сестры.

Таблица 1			Таблица 2	
ID	Фамилия_И.О.	Пол	ID_Родителя	ID_Ребенка
28	Леоненко М.Д.	Ж	38	39
38	Грант А.Б	М	58	39
39	Грант Б.А.	М	39	40
40	Грант Б.Б.	М	78	40
48	Степаненко З.А.	Ж	38	48
49	Степаненко Т.Л.	Ж	58	48
50	Степаненко Л.Л.	М	48	49
58	Кривец Д.Н.	Ж	50	49

59	Кривец Р.Д.	М	28	50
60	Колос Е.Л.	Ж	48	60
61	Колос А.Е.	М	50	60
68	Кроха М.Б.	Ж	39	68
78	Волченко Т.Д.	Ж	78	68

Пояснение.

Из первой таблицы ясно, что ID Степаненко З. А. равен 48. Найдем во второй таблице в графе «ID_ребенка» номер Степаненко З. А. Видно, что его родители имеют ID 38 и 58. Дети обладателей этих ID имеют ID 39 и 48. Обладатель ID 39 имеет детей, ID которых 40 и 68. Поскольку мы ищем племянницу, проверим пол кадого ID: 40 — М, 68 — Ж. Из первой таблицы находим, что ID 68 соответствует Кроха М. Б.

Ваш ответ: *нет ответа*. Правильный ответ: 68

↑ Задание 5 № 9185 тип 5

Для кодирования некоторой последовательности, состоящей из букв И, К, Л, М, Н, решили использовать неравномерный двоичный код, удовлетворяющий условию Фано. Для буквы Н использовали кодовое слово 0, для буквы К — кодовое слово 10. Какова наименьшая возможная суммарная длина всех пяти кодовых слов?

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

Пояснение.

Нельзя использовать кодовые слова, которые начинаются с 0 или с 10. 11 также не можем использовать, поскольку тогда мы больше не сможем взять никакое другое кодовое слово, а нам их нужно пять. Поэтому берём трёхзначное 110. 111 опять же не можем использовать, потому что понадобится ещё одно кодовое слово, а вместе с этим не останется больше свободных. Теперь осталось взять всего два слова и это будут 1110 и 1111. Итого имеем 0, 10, 110, 1110 и 1111 — 14 символов.

Ответ: 14.

Ваш ответ: *нет ответа*. Правильный ответ: 14

↑ Задание 6 № 13536 тип 6

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

1. Складываются первая и вторая, а также третья и четвёртая цифры.

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

Пример. Исходное число: 7511. Суммы: $7 + 5 = 12$; $1 + 1 = 2$. Результат: 212. Сколько существует чисел, в результате обработки которых автомат выдаст число 414

Пояснение.

Сумма 2 цифр меньше 19, значит, число 414 получается, если одна пара цифр в сумме даёт 4, а вторая — 14.

Существует два варианта получения числа 4: $3 + 1$; $1 + 3$.

Существует три варианта получения числа 14: $5 + 9$; $9 + 5$; $7 + 7$.

Можно поменять эти пары местами, поскольку они запишутся в порядке неубывания: $2 \cdot 2 \cdot 3 = 12$.

Примечание. Нужно использовать только нечетные цифры. Поэтому нельзя представить число 14 как $8 + 6$ или $6 + 8$.

Ответ: 12.

Ваш ответ: *нет ответа*. Правильный ответ: 12

↑ **Задание 7 № 7251 тип 7**

Дан фрагмент электронной таблицы:

	A	B	C	D
1	1	2	3	
2	4	5	6	
3	7	8	9	

В ячейку D1 введена формула $=A\$1*B1+C2$, а затем скопирована в ячейку D2. Какое значение в результате появится в ячейке D2?

Пояснение.

Часть формулы $A\$1$ не меняется.

$B1$ и $C2$ меняется и номер строки, и номер столбца.

Так как формулу скопировали из ячейки D1 в D2, то номер столбца не изменится, а номер строки увеличится на 1. Т. е. $B1$ становится $B2$, а $C2$ становится $C3$. Таким образом, при копировании формулы в ячейку D2, она будет иметь вид $=A\$1*B2+C3$. Подставив значения получаем: $1*5+9$.

Ваш ответ: *нет ответа*. Правильный ответ: 14

↑ **Задание 8 № 13565 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 = 1 WHILE S < 165 S = S + 15 N = N * 2 WEND PRINT N </pre>	<pre> s = 0 n = 1 while s < 165: s = s + 15 n = n * 2 print(n) </pre>
Паскаль	Алгоритмический язык
<pre> var s, n: integer; begin s := 0; n := 1; while s < 165 do begin s := s + 15; n := n * 2; end; writeln(n) end. </pre>	<pre> алг нач цел n, s n := 1 s := 0 нц пока s < 165 s = s + 15 n = n * 2 кц вывод n кон </pre>
Си++	
<pre> #include <iostream> using namespace std; int main() { </pre>	

```

int s = 0;
int n = 1;
while (s < 165) {
    s = s + 15;
    n = n * 2;
}
cout << n << endl;
return 0;
}

```

Пояснение.

Цикл `while` выполняется до тех пор, пока истинно условие $s < 165$, т. е. переменная s определяет, сколько раз выполнится цикл.

Цикл выполнится $\left\lceil \frac{165}{15} \right\rceil = 11$ раз. На одиннадцатом шаге s станет равной 165, и условие $s < 165$ окажется невыполненным, цикл прервется. Следовательно, значение n будет равно $2^{11} = 2048$.

Ответ: 2048.

Ваш ответ: *нет ответа*. Правильный ответ: 2048

↑ Задание 9 № 2428 тип 9

У Васи есть доступ к Интернет по высокоскоростному одностороннему радиоканалу, обеспечивающему скорость получения им информации 2^{19} бит в секунду. У Пети нет скоростного доступа в Интернет, но есть возможность получать информацию от Васи по низкоскоростному телефонному каналу со средней скоростью 2^{15} бит в секунду. Петя договорился с Васей, что тот будет скачивать для него данные объемом 10 Мбайт по высокоскоростному каналу и ретранслировать их Пете по низкоскоростному каналу. Компьютер Васи может начать ретрансляцию данных не раньше, чем им будут получены первые 1024 Кбайт этих данных. Каков минимально возможный промежуток времени (в секундах), с момента начала скачивания Васей данных, до полного их получения Петей? В ответе укажите только число, слово «секунд» или букву «с» добавлять не нужно.

Пояснение.

Нужно определить, сколько времени будет передаваться файл объемом 10 Мбайт по каналу со скоростью передачи данных 2^{15} бит/с; к этому времени нужно добавить задержку файла у Васи (пока он не получит 1024 Кбайт данных по каналу со скоростью 2^{19} бит/с).

Переведём объём информации в Мб в биты: $Q = 10 \text{ Мб} = 10 * 2^{20} \text{ байт} = 10 * 2^{23} \text{ бит}$.

Время задержки: $t_0 = 1024 \text{ кб} / 2^{19} \text{ бит/с} = 2^{(10 + 10 + 3) - 19} \text{ с} = 2^4 \text{ с}$.

Время скчивания данных Петей: $t_1 = 10 * 2^{23} \text{ бит} / 2^{15} \text{ бит/с} = 10 * 2^8 \text{ с}$.

Полное время: $t = t_0 + t_1 = 10 * 2^8 \text{ с} + 2^4 \text{ с} = (256 * 10 + 16) \text{ с} = 2576 \text{ с}$.

Ответ: 2576.

Ваш ответ: *нет ответа*. Правильный ответ: 2576

↑ Задание 10 № 14225 тип 10

Олег составляет таблицу кодовых слов для передачи сообщений, каждому сообщению соответствует своё кодовое слово. В качестве кодовых слов Олег использует 4-буквенные слова, в которых есть только буквы А, В, С, D, E, X, Z, причём буквы X и Z встречаются только на двух первых позициях, а буквы А, В, С, D, E — только на двух последних. Сколько различных кодовых слов может использовать Олег?

Пояснение.

Составляем четырехбуквенные слова. На первые два места можно поставить одну из двух букв X или Z. Это можно сделать $2 \cdot 2 = 4$ вариантами. На два последних места выбираем букву из 5 букв. Получаем $5 \cdot 5 = 25$ вариантов.

Таким образом, всего $25 \cdot 4 = 100$ вариантов.

Ответ: 100.

Ваш ответ: *нет ответа*. Правильный ответ: 100

↑ Задание 11 № 4650 тип 11

Последовательность чисел трибоначчи задается рекуррентным соотношением:

$$F(1) = 0$$

$$F(2) = 1$$

$$F(3) = 1$$

$$F(n) = F(n-3) + F(n-2) + F(n-1), \text{ при } n > 3, \text{ где } n - \text{ натуральное число.}$$

Чему равно девятое число в последовательности трибоначчи?

В ответе запишите только натуральное число.

Пояснение.

Последовательно находим:

$$F(4) = F(1) + F(2) + F(3) = 2,$$

$$F(5) = F(2) + F(3) + F(4) = 4,$$

$$F(6) = F(3) + F(4) + F(5) = 7,$$

$$F(7) = F(4) + F(5) + F(6) = 13,$$

$$F(8) = F(5) + F(6) + F(7) = 24,$$

$$F(9) = F(6) + F(7) + F(8) = 44.$$

Девятое число в последовательности трибоначчи равно 44.

Ваш ответ: *нет ответа*. Правильный ответ: 44

↑ Задание 12 № 2234 тип 12

Маской подсети называется 32-разрядное двоичное число, которое определяет, какая часть IP-адреса компьютера относится к адресу сети, а какая часть IP-адреса определяет адрес компьютера в подсети. В маске подсети старшие биты, отведенные в IP-адресе компьютера для адреса сети, имеют значение 1; младшие биты, отведенные в IP-адресе компьютера для адреса компьютера в подсети, имеют значение 0.

Если маска подсети 255.255.248.0 и IP-адрес компьютера в сети 112.154.133.208, то номер компьютера в сети равен _____

Пояснение.

1. Так как первые два октета (октет - число маски, содержит 8 бит) оба равны 255, то в двоичном виде они записываются как 16 единиц, а значит, первые два октета определяют адрес сети.

2. Запишем число 248 в двоичном виде.

$$248_{10} = 11111000_2$$

Итого, последние два октета маски записываются как 11111000 00000000

3. Запишем последние два октета IP-адреса компьютера в сети:

$$133_{10} = 10000101_2$$

$$208_{10} = 11010000_2$$

Итого, последние два октета IP-адреса компьютера в сети записываются так: 10000101 11010000

4. Сопоставим последние октеты маски и адреса компьютера в сети:

11111000 00000000

10000101 11010000

Жирным выделена нужная нам часть. Переведем её в десятичную систему счисления:

$$10111010000_2 = 1488_{10}.$$

Ваш ответ: *нет ответа*. Правильный ответ: 1488

↑ Задание 13 № 235 тип 13

В лыжном кроссе участвуют 777 спортсменов. Специальное устройство регистрирует прохождение каждым из участников промежуточного финиша, записывая его номер с использованием минимально возможного количества битов, одинакового для каждого спортсмена. Каков информационный объём сообщения, записанного устройством, после того как промежуточный финиш прошли 280 лыжников? (Ответ дайте в байтах.)

Пояснение.

Известно, что с помощью N бит можно закодировать 2^N различных чисел. Поскольку $2^9 < 777 < 2^{10}$ и для каждого спортсмена число бит одинаково, то для записи каждого из 777 номеров необходимо 10 бит памяти. Поскольку промежуточный финиш прошли 280 лыжников, то информационный объём сообщения составит $280 * 10 \text{ бит} = 2800 \text{ бит} = 350 \text{ байт}$.

Ваш ответ: *нет ответа*. Правильный ответ: 350

↑ Задание 14 № 1818 тип 14

Исполнитель МАШИНКА «живет» в ограниченном прямоугольном лабиринте на клетчатой плоскости, изображенном на рисунке. Серые клетки — возведенные стены, светлые — свободные клетки, по которым МАШИНКА может свободно передвигаться. По краю поля лабиринта также стоит возведенная стенка с нанесенными номерами и буквами для идентификации клеток в лабиринте.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1																1
2																2
3																3
4																4
5																5
6																6
7																7
8																8
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	

Система команд исполнителя МАШИНКА:

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

При выполнении любой из этих команд МАШИНКА перемещается на одну клетку соответственно (по отношению к наблюдателю): вверх ↑, вниз ↓, влево ←, вправо →.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится МАШИНКА (также по отношению к наблюдателю):

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

Цикл

ПОКА <условие> команда

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

При попытке передвижения на любую серую клетку МАШИНКА разбивается о стенку.

Сколько клеток приведенного лабиринта соответствуют требованию, что, стартовав в ней и выполнив предложенную ниже программу, МАШИНКА не разобьется?

НАЧАЛО

ПОКА <снизу свободно> вниз

ПОКА <слева свободно> влево

вверх

вправо

КОНЕЦ

Пояснение.

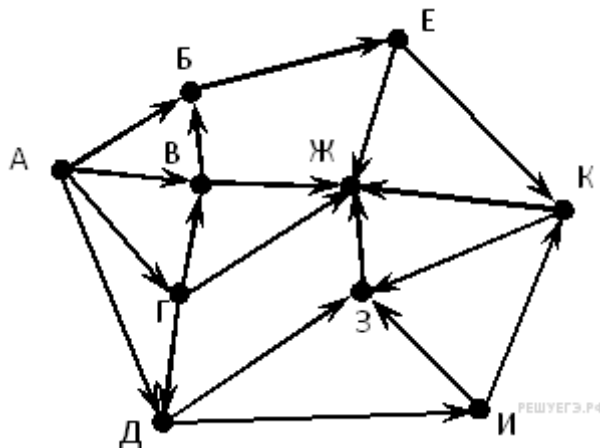
Разбиться Машинка может только при выполнении команд "вверх" и "вправо". Начав из любой клетки столбцов А, Г, Н, I, O, и клеток В2–F2, D4, J4–J6, K5–K6, L6, L4, машинка разобьётся, выполняя команду вправо. Стартовав из клеток В4–В6, С4–С5, N4–N6, M4–M5 машинка не разобьётся.

Ваш ответ: *нет ответа*. Правильный ответ: 10

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1																1
2																2
3																3
4																4
5																5
6																6
7																7
8																8
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	

↑ Задание 15 № 3291 тип 15

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



Пояснение.

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

В "Ж" можно приехать из Е, К, З, Г или В, поэтому $N = N_{Ж} = N_E + N_K + N_Z + N_G + N_B$ (1)

Аналогично:

$$N_E = N_B;$$

$$N_K = N_E + N_I;$$

$$N_Z = N_K + N_I + N_D;$$

$$N_G = N_A = 1;$$

$$N_B = N_A + N_G = 1 + 1 = 2.$$

Добавим еще вершины:

$$N_B = N_A + N_G = 1 + 1 = 2;$$

$$N_D = N_A + N_G = 1 + 1 = 2;$$

$$N_I = N_D = 2;$$

Преобразуем первые вершины с учетом значений вторых:

$$N_E = N_B = 2;$$

$$N_K = N_E + N_I = 2 + 2 = 4;$$

$$N_Z = N_K + N_I + N_D = 4 + 2 + 2 = 8;$$

$$N_G = N_A = 1;$$

$$N_B = N_A + N_G = 1 + 1 = 2.$$

Подставим в формулу (1):

$$N = N_{\text{ж}} = 3 + 5 + 9 + 1 + 2 = 20$$

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

↑ Задание 16 № 5619 тип 16

В системе счисления с некоторым основанием десятичное число 36 записывается в виде 40. Укажите это основание.

Пояснение.

Составим уравнение: $40_n = 4 \cdot n^1 + 0 \cdot n^0 = 36_{10}$, где n — основание этой системы счисления. Исходя из уравнения, $n=9$.

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

↑ Задание 17 № 4954 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)
Пекин & (Москва Токио)	338
Пекин & Москва	204
Пекин & Москва & Токио	50

Какое количество страниц (в тысячах) будет найдено по запросу *Пекин & Токио*?

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

Пояснение.

По формуле включений и исключений имеем:

$$m(\text{Пекин} \& (\text{Москва} | \text{Токио})) = m(\text{Пекин} \& \text{Москва}) + m(\text{Пекин} \& \text{Токио}) - m(\text{Пекин} \& \text{Москва} \& \text{Токио}).$$

Тогда искомое количество страниц:

$$m(\text{Пекин} \& \text{Токио}) = m(\text{Пекин} \& (\text{Москва} | \text{Токио})) - m(\text{Пекин} \& \text{Москва}) + m(\text{Пекин} \& \text{Москва} \& \text{Токио}) = 338 - 204 + 50 = 184.$$

Ваш ответ: *нет ответа*. Правильный ответ: 184

↑ Задание 18 № 13745 тип 18

Для какого наибольшего целого числа A формула

$$((x \leq 9) \rightarrow (x \cdot x \leq A)) \wedge ((y \cdot y \leq A) \rightarrow (y \leq 9))$$

тождественно истинна, то есть принимает значение 1 при любых целых неотрицательных x и y ?

Пояснение.

Раскрывая импликацию по правилу $A \rightarrow B = \neg A + B$, заменяя логическую сумму совокупностью, а логическое произведение системой соотношений, определим значения параметра A , при котором система совокупностей

$$\begin{cases} x > 9, \\ x^2 \leq A, \\ y^2 > A, \\ y \leq 9 \end{cases}$$

будет иметь решениями для любых целых неотрицательных чисел.

Заметим, что переменные не связаны между собой уравнением или неравенством, поэтому необходимо и достаточно, чтобы решениями первой совокупности были все неотрицательные x , а решениями второй совокупности были все неотрицательные y .

Решениями неравенства $x > 9$ являются числа 10, 11, 12, ... Чтобы совокупность выполнялась для всех целых неотрицательных чисел, числа 0, 1, 2, ... 9 должны быть решениями неравенства $x^2 \leq A$. Значит, $A \geq 81$.

Аналогично, решениями неравенства $y \leq 9$ являются числа 0, 1, ... 9. Следовательно, числа 10, 11, 12, ... должны быть решениями неравенства $y^2 > A$. Поэтому $A < 100$.

Тем самым, $81 \leq A < 100$. Искомое наибольшее целое значение параметра равно 99.

Ответ: 99.

Ваш ответ: *нет ответа*. Правильный ответ: 99

↑ Задание 19 № 7930 тип 19

В программе описан одномерный целочисленный массив с индексами от 0 до 10. Ниже представлен записанный на разных языках программирования фрагмент одной и той же программы, обрабатывающей данный массив.

Бейсик	Python
<pre>s = 27 n = 10 FOR i = 0 TO n-1 s = s+A(i)-A(i+1) NEXT i</pre>	<pre>s = 27 n = 10 for i in range(0,n): s = s + A[i] - A[i+1]</pre>
Паскаль	Алгоритмический язык
<pre>s:=27; n:=10; for i:=0 to n-1 do begin s:=s+A[i]-A[i+1] end;</pre>	<pre>s:=27 n:=10 нц для i от 0 до n-1 s:=s+A[i]-A[i+1] кц</pre>
Си++	
<pre>s = 27; n=10; for (i = 0; i <= n-1; i++) s=s+A[i]-A[i+1];</pre>	

Известно, что в начале выполнения этого фрагмента в массиве находилась убывающая последовательность чисел, то есть $A[0] > A[1] > \dots > A[10]$. Какое наименьшее значение может иметь переменная s после выполнения данной программы?

Пояснение.

В переменной s суммируются разности элементов, идущих друг за другом. Для того, чтобы s была наименьшей после выполнения программы, необходимо, чтобы разность элементов была наименьшей. Поскольку массив целочисленный, наименьшая разность равна единице. Алгоритм обрабатывает первые десять элементов массива, следовательно, наименьшее значение, которое может иметь переменная s после выполнения данной программы, равно $27 + 10 \cdot 1 = 37$.

Ответ: 37.

Ваш ответ: *нет ответа*. Правильный ответ: 37

↑ **Задание 20 № 4560 тип 20**

Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наименьшее из таких чисел x , при вводе которых алгоритм печатает сначала 2, а потом 21.

Бейсик	Паскаль
<pre> DIM X, A, B AS INTEGER INPUT X A=0: B=1 WHILE X > 0 A = A+1 B = B*(X MOD 10) X = X \ 10 WEND PRINT A PRINT B </pre>	<pre> var x, a, b: integer; begin readln(x); a:=0; b:=1; while x>0 do begin a:=a+1; b:=b*(x mod 10); x:= x div 10 end; writeln(a); write(b); end. </pre>
Си++	Алгоритмический язык
<pre> #include <iostream> using namespace std; int main() { int x, a, b; cin >> x; a=0; b=1; while (x>0){ a=a+1; b=b*(x%10); x= x/10; } cout << a << endl << b endl; } </pre>	<pre> алг нач цел x, a, b ввод x a:=0; b:=1 нц пока x>0 a:=a+1 b:=b*mod(x,10) x:=div(x,10) кц вывод a, нс, b кон </pre>
Python	
<pre> x = int(input()) a = 0 b = 1 while x > 0: a += 1 b *= (x % 10) x //= 10 print(a) print(b) </pre>	

Пояснение.

Рассмотрим цикл, число шагов которого зависит от изменения переменной x :
while $x > 0$ do begin

...

$x := x \text{ div } 10;$

end;

Т. к. оператор div оставляет 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 число a столько же раз увеличивается на 1. Следовательно, конечное значение a совпадает с числом цифр в x . Для того, чтобы a стало $a = 2$, x должно быть двузначным.

Теперь рассмотрим оператор изменения b :

```
while x>0 do begin
b:=b*(x mod 10);
end;
```

Оператор `mod` оставляет остаток от деления, при делении на 10 это последняя цифра x ; следовательно, число b получается произведением цифр числа x .

Представим число 21 в виде: $21 = 7 * 3$. Другого представления, в котором все множители однозначные, число 21 не имеет. Следовательно, минимальное число $x = 37$.

Ответ: 37.

Ваш ответ: *нет ответа*. Правильный ответ: 37

↑ Задание 21 № 5466 тип 21

Напишите в ответе число, которое будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках).

Бейсик	Паскаль
<pre>DIM A, B, T, M, R AS INTEGER A = -10: B = 31 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 M FUNCTION F(x) F = 2*(x*x-1)*(x*x-1)+17 END FUNCTION</pre>	<pre>var a,b,t,M,R :integer; Function F(x:integer):integer; begin F := 2*(x*x-1)*(x*x-1)+17 end; begin a := -10; b := 31; 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(M) end.</pre>
Си++	Алгоритмический
<pre>#include <iostream> using namespace std; int F(int x) { return 2*(x*x-1)*(x*x-1)+17; } int main() { int a, b, t, M, R; a = -10; b = 31; M = a; R = F(a); for (t = a; t <= b; t++) { if (F(t) > R) { M = t; R = F(t); } } cout << M << endl; }</pre>	<pre>алг нач цел a, b, t, M, R a := -10; b := 31 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 M кон алг цел F(цел x) нач знач:=2*(x*x-1)*(x*x-1)+17 кон</pre>

Python

```
def f(x):
    return 2*(x*x-1)*(x*x-1)+17
a = -10
b = 31
M = a
R = f(a)
for t in range(a, b+1):
    if (f(t) > R):
        M = t
        R = f(t);
print(M)
```

Пояснение.

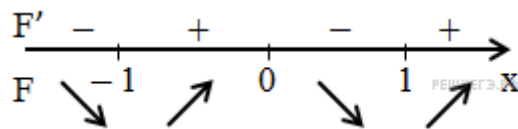
Алгоритм предназначен для поиска наибольшего t , при котором функция $F(t)$ имеет наибольшее значение на отрезке от a до b . Преобразуем функцию:

$$F(x) = 2(x^2 - 1)(x^2 - 1) = 2x^4 - 4x^2 + 2.$$

Вычислим производную функции:

$$F'(x) = 8x^3 - 8x = 8x(x^2 - 1) = 8x(x - 1)(x + 1).$$

Нули производной: $x = 0$, $x = 1$, $x = -1$. Воспользуемся методом интервалов:



Поскольку алгоритм осуществляет поиск точки t , в которой $F(t)$ принимает наибольшее значение на отрезке от -10 до 31 , то переменная M будет принимать значение 0 или 31 . Подставляя в функцию значения 0 и 31 , находим, что наибольшее значение функции достигается в точке 31 , поэтому переменной M будет присвоено значение 31 .

Ответ: 31.

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

↑ Задание 22 № 6345 тип 22

У исполнителя Удвоитель две команды, которым присвоены номера:

1. прибавь 1,
2. умножь на 2.

Первая из них увеличивает число на экране на 1, вторая удваивает его. Программа для Удвоителя — это последовательность команд. Сколько есть программ, которые число 2 преобразуют в число 23?

Пояснение.

Обозначим $R(n)$ — количество программ, которые преобразуют число 2 в число n . Обозначим $t(n)$ наибольшее кратное 2, не превосходящее n . Верны следующие соотношения:

1. Если n не делится на 2, то тогда $R(n) = R(t(n))$, так как существует единственный способ получения n из $t(n)$ — прибавлением единиц.

2. Пусть n делится на 2. Тогда $R(n) = R(n / 2) + R(n - 1) = R(n / 2) + R(n - 2)$ (если $n > 2$). Таким образом, достаточно вычислить значения $R(n)$ для всех чисел, кратных 2 и не превосходящих 23. Имеем:

$$\begin{aligned} R(2) &= 1 = R(3), \\ R(4) &= R(2) + R(3) = 1 + 1 = 2 = R(5), \end{aligned}$$

$$\begin{aligned}
 R(6) &= R(3) + R(5) = 1 + 2 = 3 = R(7), \\
 R(8) &= R(4) + R(7) = 2 + 3 = 5 = R(9), \\
 R(10) &= R(5) + R(9) = 2 + 5 = 7 = R(11), \\
 R(12) &= R(6) + R(11) = 3 + 7 = 10 = R(13), \\
 R(14) &= R(7) + R(13) = 3 + 10 = 13 = R(15), \\
 R(16) &= R(8) + R(15) = 5 + 13 = 18 = R(17), \\
 R(18) &= R(9) + R(17) = 5 + 18 = 23 = R(19), \\
 R(20) &= R(10) + R(19) = 7 + 23 = 30 = R(21), \\
 R(22) &= R(11) + R(21) = 7 + 30 = 37.
 \end{aligned}$$

Ответ: 37.

Ваш ответ: *нет ответа*. Правильный ответ: 37

↑ Задание 23 № 2201 тип 23

Сколько различных решений имеет уравнение $J \wedge \neg K \wedge L \wedge \neg M \wedge (N \vee \neg N) = 0$, где J, K, L, M, N — логические переменные?

В ответе не нужно перечислять все различные наборы значений J, K, L, M и N , при которых выполнено данное равенство. В качестве ответа нужно указать количество таких наборов.

Пояснение.

Выражение $(N \vee \neg N)$ истинно при любом N , поэтому

$$J \wedge \neg K \wedge L \wedge \neg M = 0.$$

Применим отрицание к обеим частям логического уравнения и используем закон де Моргана $\neg(A \wedge B) = \neg A \vee \neg B$. Получим

$$\neg J \vee K \vee \neg L \vee M = 1.$$

Логическая сумма равна 1, если хотя бы одно из составляющих ее высказываний равно 1. Поэтому полученному уравнению удовлетворяют любые комбинации логических переменных кроме случая, когда все входящие в уравнение величины равны 0. Каждая из 4 переменных может быть равна либо 1, либо 0, поэтому всевозможных комбинаций $2 \cdot 2 \cdot 2 \cdot 2 = 16$. Следовательно, уравнение имеет $16 - 1 = 15$ решений.

Осталось заметить, что найденные 15 решений соответствуют любому из двух возможных значений значений логической переменной N , поэтому исходное уравнение имеет 30 решений.

Ваш ответ: *нет ответа*. Правильный ответ: 30

Проверка части с развернутым ответом

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

Задание 24 (С1) № 15146

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. <ol style="list-style-type: none"> 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. <ol style="list-style-type: none"> 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 вывести минимальное натуральное число K , для которого $1! + 2! + \dots + K! > A$.

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, S AS INTEGER INPUT A K = 1 F = 1 S = 1 WHILE S <= A F = F * K K = K + 1 S = S + F WEND PRINT K END </pre>	<pre> a = int(input()) k = 1 f = 1 s = 1 while s <= a: f *= k k += 1 s += f print(k) </pre>
Паскаль	Алгоритмический язык
<pre> var a, k, f, s: integer; begin read(a); k := 1; f := 1; s := 1; while s <= a do begin f := f * k; </pre>	<pre> алг нач цел a, k, f, s ввод a k := 1 f := 1 s := 1 нц пока s <= a </pre>

<pre> k := k + 1; s := s + f end; writeln(k) end. </pre>	<pre> f := f * k; k := k + 1; s := s + f кц вывод k кон </pre>
C++	
<pre> #include <iostream> using namespace std; int main(){ int a, k, f, s; cin >> a; k = 1; f = 1; s = 1; while (s <= a) { f *= k; ++k; s += f; } cout << k; return 0; } </pre>	

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе $A = 6$.
2. Назовите **минимальное A , большее 10**, при котором программа выведет **верный** ответ.
3. Найдите в программе все ошибки (известно, что их не более двух). 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 **верный** ответ, равно 33. Программа выдаёт верный ответ 5. При вводе чисел от 11 до 32 программа выдаёт ответ 5 вместо верного ответа 4.
3. Программа содержит две ошибки.
 - 1) Неверная инициализация. Значения переменных K и S не согласованы: эти переменные могут быть равны 1 и 0 или 2 и 1, но не 1 и 1. В результате этой ошибки текущее значение суммы всегда оказывается на 1 больше правильного.
 - 2) Неверный порядок действий в цикле. В результате факториал и сумма факториалов вычисляются для предыдущего значения K . Поскольку менять строки местами не разрешается, для исправления ошибки нужно внести коррективы при выводе ответа.

Пример исправления для языка Паскаль:

Первая ошибка, способ 1:

```

k := 1;
Исправленная строка:
k := 2;

```

Первая ошибка, способ 2:

```

s := 1;
Исправленная строка:
s := 0;

```

Вторая ошибка:

```

writeln(k)
Исправленная строка:
writeln(k-1)

```


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) № 5757

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"> 1) не инициализируется или неверно инициализируется переменная MAX (например, присваивается начальное значение, большее или равное 100); 2) неточно определяется принадлежность диапазону трехзначных чисел (например, в сравнении со 100 вместо знака «больше или равно» используется знак «больше»); 3) неверно осуществляется проверка делимости на 7; 4) на делимость на 7 проверяется не значение элемента, а его индекс; 5) в сложном условии вместо логической операции «И» используется логическая операция «ИЛИ»; 6) отсутствует вывод ответа, или при выводе ответа не учитывается случай, что требуемого числа в массиве может не быть; 7) используется переменная, не объявленная в разделе описания переменных; 8) не указано или неверно указано условие завершения цикла; 9) индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно; 10) неверно расставлены операторные скобки. 	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, делящихся на 5. Если в исходном массиве нет элемента, значение которого является трёхзначным числом и при этом кратно 5, 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 <iostream> using namespace std; #define N 20</pre>	<pre>алг нач цел N = 20</pre>

<pre>int main() { int a[N]; int i, j, max; for (i = 0; i < N; i++) cin >> a[i]; ... }</pre>	<pre>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 // 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.

Пример правильного решения на языке Паскаль

```
max := 99;
for i = 1 to N do
if (a[i] >= 100) and (a[i] <= 999) and (a[i] mod 5 = 0) and
(a[i] > max) then
max := a[i];
if max > 99
then writeln(max)
else
writeln('Не найдено');
```

На алгоритмическом языке

```
если a[i] >= 100 и a[i] <= 999 и mod(a[i], 5)=0 и a[i] > max
то
max := a[i]
все
кц
если max > 99
вывод max
иначе
вывод "Не найдено"
все
```

На языке Бейсик

```
MAX = 99
FOR I = 1 TO N
IF A(I) >= 100 AND A(I) <= 999 AND A(I) MOD 5=0 AND A(I) > MAX
THEN
MAX = A(I)
END IF
NEXT I
IF MAX > 99 THEN
PRINT MAX
ELSE
PRINT "Не найдено"
END IF
```

На языке Си

```

max = 99;
for (i = 0; i < N; i++)
if (a[i] > 99 && a[i] < 1000 && a[i]%5==0 && a[i]>max)
max = a[i];
if (max>99)
cout << max << endl;
else
cout << "Не найдено";

```

На естественном языке

Записываем в переменную MAX начальное значение, равное 99. В цикле от первого элемента до двадцатого находим остаток от деления элемента исходного массива на 5. Если значение данного остатка 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»

Задание 26 (С3) № 15148

Критерии оценивания выполнения задания	Баллы
<p>Выполнены второе и третье задания.</p> <p>Для первого задания правильно перечислены позиции, в которых Паша выигрывает первым ходом (пункт 1(а)), и правильно указано, кто из игроков имеет выигрышную стратегию при указанных значениях S (пункт 1(б)). При этом допускаются недочёты следующих типов:</p> <ul style="list-style-type: none"> — в п. 1(а) не указано, каким ходом выигрывает Паша; — в п. 1(б) не указано, что игрокам нет смысла утраивать количество камней в куче. <p>Здесь и далее в решениях допускаются арифметические ошибки, которые не искажают сути решения и не приводят к неправильному ответу</p>	3
<p>Не выполнены условия, позволяющие поставить 3 балла, и выполнено одно из следующих условий:</p> <ul style="list-style-type: none"> — выполнено третье задание; — выполнены первое и второе задания; — первое задание выполнено, возможно, при наличии недочётов, указанных в критериях на 3 балла; для второго задания (i) правильно указано, кто из игроков имеет выигрышную стратегию в каждой из указанных позиций, и (ii) правильно указан первый ход Паши при выигрышной стратегии, однако не указано, что после выбранного хода Паши получается позиция, выигрышная для Вали; для третьего задания правильно указан игрок, имеющий выигрышную стратегию 	2
<p>Не выполнены условия, позволяющие поставить 3 или 2 балла, и выполнено одно из следующих условий:</p> <ul style="list-style-type: none"> — первое задание выполнено, возможно, с недочётами, указанными в критериях на 3 балла; — второе задание выполнено, возможно, с недочётами, указанными в критериях на 2 балла; — для заданий 2 и 3 во всех случаях правильно указан игрок, имеющий выигрышную стратегию 	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
добавить в кучу два камня или
увеличить количество камней в куче в два раза.**

Например, имея кучу из 10 камней, за один ход можно получить кучу из 11, 12 или 20 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче превышает 41. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 42 или

больше камней.

В начальный момент в куче было S камней, $1 \leq S \leq 41$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. В описание выигрышной стратегии не следует включать ходы следующей стратегии игрока, которые не являются для него безусловно выигрышными.

Выполните следующие задания.

Задание 1.

а) Назовите все значения S , при которых Петя может выиграть первым ходом, причём у Пети есть ровно один выигрышающий ход.

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.

Задание 2.

Укажите три значения S , при которых у Пети есть выигрышная стратегия, причём Петя не может выиграть первым ходом, но Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня. Для указанных значений S опишите выигрышную стратегию Пети.

Задание 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.

Дерево не должно содержать партий, невозможных при реализации выигрышающим игроком своей выигрышной стратегии. Например, полное дерево игры не будет верным ответом на это задание.

Пояснение.

Задание 1.

а) Петя может выиграть единственным способом (увеличив количество камней в 2 раза), если $S = 21, \dots, 39$. При меньших значениях S за один ход нельзя получить кучу, в которой будет 42 или более камней. При $S = 40$ и $S = 41$ у Пети есть более одного выигрышающего хода.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет $S = 20$ камней. Тогда после первого хода Пети в куче будет 21, 22 или 40 камней. Во всех случаях Ваня увеличивает количество камней в 2 раза и выигрывает в один ход.

Задание 2.

Возможные значения S : 10, 18, 19. В этих случаях Петя не может выиграть первым ходом. Однако он может получить кучу из 20 камней (при $S = 10$ он удваивает количество камней; при $S = 18$ добавляет 2 камня; при $S = 19$ добавляет один камень). Эта позиция разобрана в п. 16. В ней игрок, который будет ходить (в данном случае это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.

Задание 3.

Возможное значение S : 17. После первого хода Пети в куче будет 18, 19 или 34 камня. Если в куче станет 34 камня, то Ваня увеличит количество камней в 2 раза и выигрывает своим первым ходом. Ситуация, когда в куче 18 или 19 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).

Исходное положение	1-й ход Пети (разобраны все ходы, указана полученная позиция)	1-й ход Вани (только ход по стратегии, указана полученная позиция)	2-й ход Пети (разобраны все ходы, указана полученная позиция)	2-й ход Вани (только ход по стратегии, указана полученная позиция)
17	$17 + 1 = 18$	$18 + 2 = 20$	$20 + 1 = 21$	$21 * 2 = 42 >>$
			$20 + 2 = 22$	$22 * 2 = 44 >>$
			$20 * 2 = 40$	$40 * 2 = 80 >>$
	$17 + 2 = 19$	$19 + 1 = 20$	$20 + 1 = 21$	$21 * 2 = 42 >>$
			$20 + 2 = 22$	$22 * 2 = 44 >>$
			$20 * 2 = 40$	$40 * 2 = 80 >>$
$17 * 2 = 34$	$34 * 2 = 68 >>$			

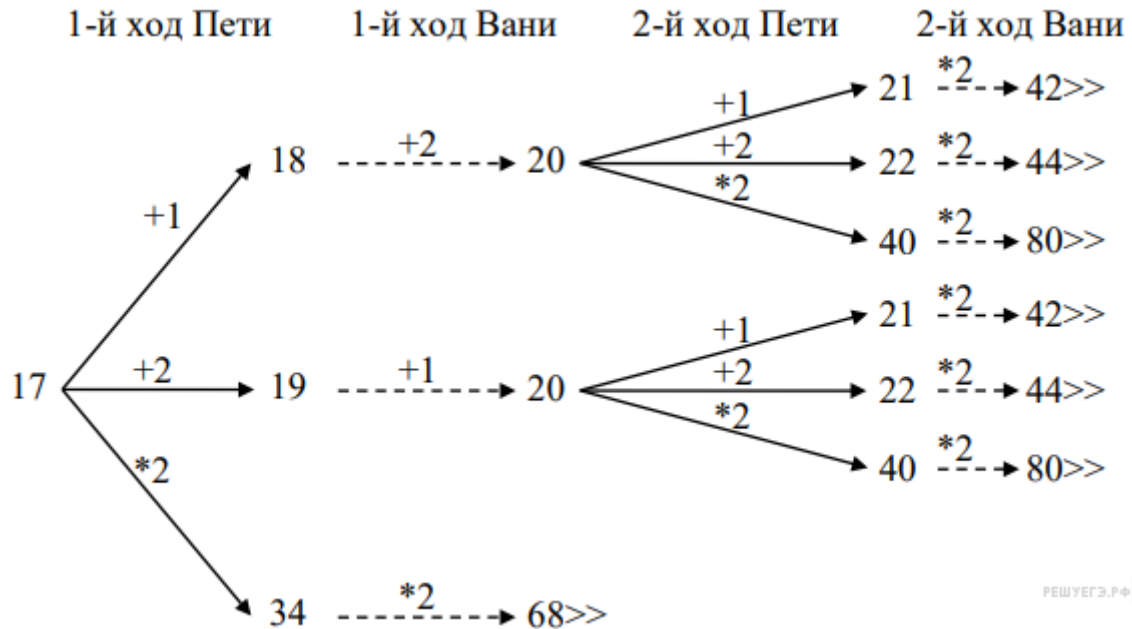


Рис. 1. Дерево всех партий, возможных при описанной стратегии Пети. Ходы Пети показаны сплошными стрелками, ходы Вани показаны пунктирными стрелками. Заключительные позиции обозначены знаком >>.

Задание 27 (С4) № 15839

Критерии оценивания выполнения задания	Баллы
<p>Программа правильно работает для любых входных данных произвольного размера. Используемая память не зависит от количества прочитанных чисел, а время работы пропорционально этому количеству.</p> <p>Допускается наличие в тексте программы до трёх синтаксических ошибок одного из следующих видов:</p> <ol style="list-style-type: none"> 1) пропущен или неверно указан знак пунктуации, 2) неверно написано или пропущено зарезервированное слово языка программирования, 3) не описана или неверно описана переменная, 4) применяется операция, недопустимая для соответствующего типа данных. <p>Если одна и та же ошибка встречается несколько раз, это считается за одну ошибку</p>	4
<p>Не выполнены условия, позволяющие поставить 4 балла.</p> <p>Программа в целом работает правильно для любых входных данных произвольного размера. Время работы пропорционально количеству введённых чисел, правильно указано, какие величины должны вычисляться по ходу чтения элементов последовательности чисел.</p> <p>Используемая память, возможно, зависит от количества прочитанных чисел (например, входные данные запоминаются в массиве, контейнере STL в C++ или другой аналогичной структуре данных).</p> <p>Количество синтаксических ошибок («описок»), указанных в критериях на 4 балла, — не более пяти.</p> <p>Допускается наличие не более одной ошибки следующих видов:</p> <ol style="list-style-type: none"> 1) ошибка при инициализации суммы и/или минимума; 2) неверно определяется результат в ситуации, когда все числа кратны 8; 3) допущен выход за границу массива; 4) используется знак < вместо <=, or вместо and и т. п.; 5) неверный формат вывода (выведено одно число вместо двух или числа при выводе поменялись местами) 	3
<p>Не выполнены условия, позволяющие поставить 3 или 4 балла.</p> <p>Программа работает в целом верно, эффективно или нет, но в реализации алгоритма есть до трёх содержательных ошибок из следующего списка:</p> <ol style="list-style-type: none"> 1)–5) см. список в критериях на 3 балла; 6) вместо минимального значения, не кратного 8, ищется абсолютное минимальное значение. <p>Количество синтаксических ошибок, указанных в критериях на 4 балла, не должно быть более девяти</p>	2
<p>Не выполнены условия, позволяющие поставить 2, 3 или 4 балла.</p>	1

Программа работает правильно в отдельных частных случаях. Например, все входные данные сохраняются в массиве, и делается попытка перебором найти наилучшую комбинацию. Допускается любое количество синтаксических ошибок	
Не выполнены критерии, позволяющие поставить 1, 2, 3 или 4 балла	0
<i>Максимальный балл</i>	4

Дан набор из N целых положительных чисел. Из этих чисел формируются все возможные пары (парой считаются два элемента, которые находятся на разных местах в наборе, порядок чисел в паре не учитывается), в каждой паре вычисляются сумма и произведение элементов. Необходимо определить количество пар, у которых сумма нечётна, а произведение делится на 5.

Напишите эффективную по времени и по памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества исходных чисел N в k раз время работы программы увеличивается не более чем в k раз.

Программа считается эффективной по памяти, если память, необходимая для хранения всех переменных программы, не превышает одного килобайта и не увеличивается с ростом N .

Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную только по времени или только по памяти, — 3 балла.

Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, — 2 балла.

Вы можете сдать одну или две программы решения задачи. Если Вы сдадите две программы, каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

Перед текстом программы кратко опишите алгоритм решения. Укажите использованный язык программирования и его версию.

Описание входных и выходных данных.

В первой строке входных данных задаётся количество чисел N ($1 \leq N \leq 1000$). В каждой из последующих N строк записано одно натуральное число, не превышающее 100.

Пример входных данных:

```
4
1
2
4
5
```

Пример выходных данных для приведённого выше примера входных данных:

```
2
```

Из четырёх чисел можно составить 6 пар. В данном случае условиям удовлетворяют две пары: (2, 5) и (4, 5). Суммы чисел в этих парах (7 и 9) нечётны, а произведения (10 и 20) делятся на 5. У всех остальных пар как минимум одно из этих условий не выполняется.

Пояснение.

Чтобы сумма двух чисел было нечётной, одно из них должно быть чётным, а другое — нечётным. Чтобы произведение двух чисел делилось на 5, хотя бы одно из этих чисел должно делиться на 5.

Разобьём все числа исходного набора на 4 группы, в зависимости от их чётности и делимости на 5. Сами числа можно не хранить, достаточно при вводе определить остаток от деления очередного числа на 2 и на 5 и увеличить соответствующий счётчик. Таким образом, независимо от количества чисел в исходном наборе, после чтения исходных данных для хранения необходимой информации хватит четырёх счётчиков, и программа получится эффективной по памяти.

Пусть в результате подсчёта получилось, что в наборе данных

m_1 нечётных чисел, не кратных 5;

m_2 чётных чисел, не кратных 5;

m_5 нечётных чисел, кратных 5;

m_{10} чётных чисел, кратных 5.

Тогда количество удовлетворяющих условию пар можно определить по формуле $m_1m_{10} + m_2m_5 + m_5m_{10}$.

Ниже приведена реализующая описанный выше алгоритм программа на языке Паскаль (использована версия PascalABC).

Пример правильной и эффективной программы на языке Паскаль:

```
var
  N: integer; {количество чисел}
  a: integer; {очередное число}
```

```

m1: integer; {нечётные, не кратные 5}
m2: integer; {чётные, не кратные 5}
m5: integer; {нечётные, кратные 5}
m10: integer; {чётные, кратные 5}
s: integer; {количество пар}
i: integer;
begin
  m1 := 0; m2 := 0; m5 := 0; m10 := 0;
  readln(N);
  for i:=0 to N do begin
    readln(a);
    if a mod 2 = 0 then begin
      if a mod 5 = 0 then m10 := m10 + 1;
      else m2 := m2 + 1;
    end
    else begin
      if a mod 5 = 0 then m5 := m5 + 1;
      else m1 := m1 + 1;
    end
  end;
  s := m1 * m10 + m2 * m5 + m5 * m10;
  writeln(s);
end.

```

Возможно также «лобовое» решение: запишем все исходные числа в массив, переберём все возможные пары и подсчитаем количество подходящих. Такое решение не является эффективным ни по памяти (требуемая память зависит от размера исходных данных), ни по времени (количество возможных пар, а значит, количество действий и время счёта с ростом количества исходных элементов растёт квадратично). Такая программа оценивается не выше двух баллов.

Ниже приведена реализующая описанный выше алгоритм программа на языке Паскаль (использована версия PascalABC)

Пример правильной, но неэффективной программы на языке Паскаль

```

var
  N: integer; {количество чисел}
  a: array [1..1000] of integer; {исходные данные}
  s: integer; {количество пар}
  i, j: integer;
begin
  readln(N);
  for i := 1 to N do readln(a[i]);
  s := 0;
  for i := 1 to N - 1 do begin
    for j := i + 1 to N do begin
      if ((a[i] + a[j]) mod 2 = 1) and ((a[i]*a[j]) mod 5 = 0) then s := s + 1;
    end
  end;
  writeln(s);
end.

```