

Решения

↑ Задание 1 № 6907 тип 1

Укажите целое число от 7 до 10, двоичная запись которого содержит ровно два значащих нуля. Если таких чисел несколько, укажите наибольшее из них.

Пояснение.

Представим все числа в двоичной системе счисления:

$$\begin{aligned} 7_{10} &= 111_2, \\ 8_{10} &= 1000_2, \\ 9_{10} &= 1001_2, \\ 10_{10} &= 1010_2. \end{aligned}$$

Из чисел 9 и 10 выбираем число 10, поскольку оно является наибольшим.

Ответ: 10.

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

↑ Задание 2 № 14688 тип 2

Логическая функция F задаётся выражением $(x \vee y) \rightarrow (z \equiv x)$.

Дан частично заполненный фрагмент, содержащий **неповторяющиеся** строки таблицы истинности функции F .

Определите, какому столбцу таблицы истинности соответствует каждая из переменных x, y, z .

| Переменная 1 | Переменная 2 | Переменная 3 | Функция |
|--------------|--------------|--------------|---------|
| ??? | ??? | ??? | F |
| | 0 | 0 | 0 |
| | 0 | | 0 |

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 фрагмент таблицы истинности:

| Переменная 1 | Переменная 2 | Функция |
|--------------|--------------|---------|
| ??? | ??? | F |
| 0 | 1 | 0 |

Тогда первому столбцу соответствует переменная y , а второму столбцу соответствует переменная x . В ответе нужно написать: yx .

Пояснение.

Данная импликация принимает значение 0 тогда и только тогда, когда $\begin{cases} x + y = 1, \\ x \neq z. \end{cases} \quad (*)$

Пусть $x = 0$, тогда $y = z = 1$. В первой строке нет двух единиц, значит, $x = 1$, и эта переменная находится в первом столбце. Тогда первая строка имеет вид 1 0 0.

Вторая строка должна отличаться от первой, поэтому она имеет вид 1 0 1. Рассмотрим два варианта:

| | | |
|---|---|---|
| x | y | z |
| 1 | 0 | 0 |
| 1 | 0 | 1 |

| | | |
|---|---|---|
| x | z | y |
| 1 | 0 | 0 |
| 1 | 0 | 1 |

Первый вариант не удовлетворяет системе (*), а второй удовлетворяет.

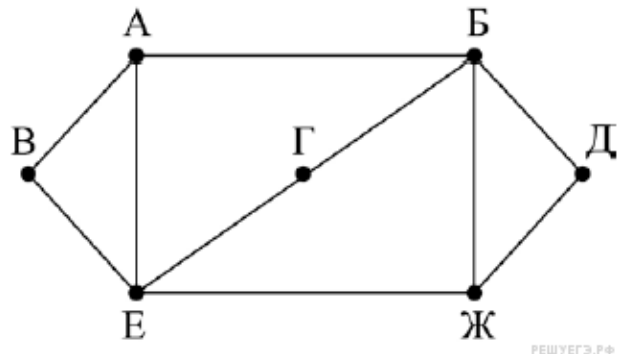
Ответ: хзу.

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

↑ Задание 3 № 16806 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 | | 7 | | | | | 5 |
| П2 | 7 | | 11 | | | | 12 |
| П3 | | 11 | | 6 | 10 | 8 | |
| П4 | | | 6 | | 9 | | |
| П5 | | | 10 | 9 | | | 9 |
| П6 | | | 8 | | | | 7 |
| П7 | 5 | 12 | | | 9 | 7 | |



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 П6. Далее рассмотрим два варианта:

1. Е соответствует П3, а Б соответствует П7. Значит, В соответствует П4, а А соответствует П5. Тогда Д соответствует П1, а Ж соответствует П2. Тогда кратчайшее расстояние из пункта В в пункт Д равняется 23.

2. Е соответствует П7, а Б соответствует П3. Значит, В соответствует П1, а А соответствует П2. Тогда Д соответствует П4, а Ж соответствует П5. Тогда кратчайшее расстояние из пункта В в пункт Д равняется 23.

Ответ: 23.

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

↑ Задание 4 № 14219 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. На основании приведённых данных определите, сколько всего внуков и внучек у Кравец Д.К.

| Таблица 1 | | | Таблица 2 | |
|-----------|--------------|-----|-------------|------------|
| ID | Фамилия_И.О. | Пол | ID_Родителя | ID_Ребенка |
| 866 | Кравец Д.К. | Ж | 866 | 911 |
| 867 | Тошич Б.Ф. | М | 866 | 938 |
| 879 | Гонтарь В.А. | Ж | 867 | 911 |

| | | | | |
|------|-------------|-----|------|------|
| 885 | Крон К.Г. | М | 867 | 938 |
| 900 | Кислюк Л.А. | Ж | 911 | 879 |
| 904 | Петраш А.И. | М | 911 | 1041 |
| 911 | Тошич А.Б. | Ж | 904 | 900 |
| 932 | Петраш П.А. | Ж | 938 | 995 |
| 938 | Тошич И.Б. | М | 938 | 1017 |
| 949 | Седых Г.Р. | Ж | 949 | 995 |
| 970 | Кислюк А.П. | М | 949 | 1017 |
| 995 | Тошич Т.И. | Ж | 970 | 879 |
| 1017 | Тошич П.И. | М | 970 | 1041 |
| 1026 | Мухина Р.Г. | Ж | 904 | 932 |
| 1041 | Сайко М.А. | Ж | 1026 | 900 |
| 1056 | Кислюк П.А. | М | 1026 | 932 |
| ... | ... | ... | ... | ... |

Пояснение.

Из первой таблицы определяем, что id Кравец Д. К. — 866.

Из второй определяем, что такому id соответствуют id 911 и 938.

Из второй определяем, что id 911 соответствуют id 879, 1041, а id 938 — 995 и 1017.

Всего у Кравец Д. К. 4 внуков с id 879, 1041, 995 и 1017.

Ответ: 4.

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

↑ Задание 5 № 7746 тип 5

Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, используется неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Вот этот код: А — 1; Б — 0100; В — 000; Г — 011; Д — 0101. Требуется сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было декодировать однозначно. Коды остальных букв меняться не должны. Каким из указанных способов это можно сделать?

- 1) для буквы Г — 11
- 2) для буквы В — 00
- 3) для буквы Г — 01
- 4) это невозможно

Пояснение.

Для однозначного декодирования получившееся в результате сокращения кодовое слово не должно быть началом никакого другого. Первый вариант ответа не подходит, поскольку код буквы А является началом кода буквы Г. Второй вариант ответа подходит. Третий вариант ответа не подходит, т. к. в таком случае код буквы Г является началом кода буквы Д.

Правильный ответ указан под номером: 2.

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

↑ Задание 6 № 15818 тип 6

Автомат обрабатывает натуральное число N по следующему алгоритму:

1. Строится двоичная запись числа N .
2. Складываются все цифры полученной двоичной записи. В конец записи (справа) дописывается остаток от деления суммы на 2.
3. Предыдущий пункт повторяется для записи с добавленной цифрой.

4. Результат переводится в десятичную систему и выводится на экран.

Пример. Дано число $N = 13$. Алгоритм работает следующим образом:

1. Двоичная запись числа N : 1101.
2. Сумма цифр двоичной записи 3, остаток от деления на 2 равен 1, новая запись 11011.
3. Сумма цифр полученной записи 4, остаток от деления на 2 равен 0, новая запись 110110.
4. На экран выводится число 54.

Какое наименьшее число, большее 93, может появиться на экране в результате работы автомата?

Пояснение.

Рассмотрим числа, большие 93, и найдем меньшее число, которое является результатом работы алгоритма.

$94 = 1011110_2$ — не является результатом работы алгоритма.

$95 = 1011111_2$ — не является результатом работы алгоритма.

$96 = 1100000_2$ — является результатом работы алгоритма для числа 11000_2 .

Ответ: 96.

Ваш ответ: *нет ответа*. Правильный ответ: 96

↑ Задание 7 № 7478 тип 7

Дан фрагмент электронной таблицы.

| | A | B | C |
|---|-----------------------|----------------------|-----------------|
| 1 | | 4 | 6 |
| 2 | $= (A1 - 2)/(B1 - 1)$ | $= C1*B1/(4*A1 + 4)$ | $= C1/(A1 - 2)$ |



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

Пояснение.

Преобразуем таблицу.

| | A | B | C |
|---|----------------|-------------------|----------------|
| 1 | | 4 | 6 |
| 2 | $= (A1 - 2)/3$ | $= 24/(4*A1 + 4)$ | $= 6/(A1 - 2)$ |



Из диаграммы видно, что значения в двух ячейках равны, а значение в третьей в два раза больше. Приравняем значения в ячейках A2 и B2: $2 \cdot (A1 - 2)/3 = 6/(A1 + 1)$, откуда $A1 = 5$. Подставив найденное значение $A1 = 5$, убеждаемся, что значения во всех ячейках имеют один и тот же знак.

Ответ: 5.

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

↑ Задание 8 № 6458 тип 8

Определите число, которое будет напечатано в результате выполнения программы (записанной ниже на разных языках программирования).

| Бейсик | Паскаль |
|--|---|
| <pre> DIM N, S AS INTEGER N = 20 S = 0 WHILE S <= 257 S = S + 10 N = N + 3 </pre> | <pre> var n, s: integer; begin n := 20; s := 0; while s <= 257 do begin </pre> |

| | |
|---|--|
| <pre>WEND PRINT N</pre> | <pre>s := s + 10; n := n + 3 end; write(n) end.</pre> |
| Си++ | Алгоритмический язык |
| <pre>#include <iostream> using namespace std; int main() { int n, s; n = 20; s = 0; while (s <= 257) { s = s + 10; n = n + 3; } cout << n << endl; }</pre> | <pre>алг нач цел n, s n := 20 s := 0 нц пока s <= 257 s := s + 10 n := n + 3 кц вывод n кон</pre> |
| Python | |
| <pre>n = 20 s = 0 while s <= 257: s += 10 n += 3 print(n)</pre> | |

Пояснение.

Цикл while выполняется до тех пор, пока истинно условие $s \leq 257$, т. е. переменная s определяет, сколько раз выполнится цикл.

Заметим, что $\left\lceil \frac{257}{10} \right\rceil = 26$. На 26 шаге s станет равной 260 и условие $s \leq 257$ окажется невыполненным, цикл прервется. Всего 26 шагов, следовательно, значение n будет равно $20 + 26 \cdot 3 = 98$.

Ответ: 98.

Ваш ответ: *нет ответа*. Правильный ответ: 98

↑ Задание 9 № 6897 тип 9

Документ объёмом 16 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^{20} бит в секунду;
- объём сжатого архиватором документа равен 25% 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: $16 \cdot 0,25 = 4$ Мбайт. Переведём Q из Мбайт в биты: $4 \text{ Мбайт} = 4 \cdot 2^{20} \text{ байт} = 4 \cdot 2^{23} \text{ бит}$. Найдём общее время: $t = 18 \text{ с} + 2 \text{ с} + 4 \cdot 2^{23} \text{ бит} / 2^{20} \text{ бит/с} = 20 + 32 \text{ с} =$

52 с.

Способ Б. Общее время совпадает с временем передачи: $t = 16 \cdot 2^{23} \text{ бит} / 2^{20} \text{ бит/с} = 128 \text{ с}$.
Таким образом, способ А быстрее на $128 - 52 = 76 \text{ с}$.

Ответ: А76.

Ваш ответ: *нет ответа*. Правильный ответ: А76

↑ Задание 10 № 4790 тип 10

Шахматная доска состоит из 8 столбцов и 8 строк. Какое минимальное количество бит потребуется для кодирования координат одной шахматной клетки?

Пояснение.

Если в алфавите M символов, то количество всех возможных «слов» (сообщений) длиной N равно $Q = M^N$.

Всего клеток $Q = 64$. В алфавите 2 символа (так как «бит»), то есть $M = 2$. Осталось найти N .
 $64 = 2^N$, следовательно, $N = 6$.

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

↑ Задание 11 № 4656 тип 11

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

$$F(1) = 0$$

$$F(n) = F(n-1) + n, \text{ при } n > 1$$

$$G(1) = 1$$

$$G(n) = G(n-1) * n, \text{ при } n > 1$$

Чему равно значение функции $F(5) + G(5)$?

В ответе запишите только натуральное число.

Пояснение.

Последовательно находим:

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

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

$$F(4) = F(3) + 4 = 9,$$

$$F(5) = F(4) + 5 = 14,$$

$$G(2) = G(1) * 2 = 2,$$

$$G(3) = G(2) * 3 = 6,$$

$$G(4) = G(3) * 4 = 24,$$

$$G(5) = G(4) * 5 = 120.$$

Затем находим $F(5) + G(5) = 14 + 120 = 134$.

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

↑ Задание 12 № 6784 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 сети и маске определите адрес сети.

IP-адрес узла: 224.24.254.134

Маска: 255.255.240.0

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

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|

| | | | | | | | |
|-----|-----|-----|-----|-----|----|---|---|
| 255 | 240 | 232 | 224 | 234 | 24 | 8 | 0 |
|-----|-----|-----|-----|-----|----|---|---|

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

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

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

Пояснение.

1. Запишем числа маски сети в двоичной системе счисления:

$$255_{10} = 11111111_2, \quad 240_{10} = 11110000_2, \quad 0_{10} = 00000000_2.$$

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 240 с числом 254:

$$240_{10} = 11110000_2, \quad 254_{10} = 11111110_2.$$

Результатом конъюнкции является число $11110000_2 = 240_{10}$.

4. Сопоставим варианты ответа получившимся числам: 224, 24, 240, 0.

Таким образом, ответ: DFBH.

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

↑ Задание 13 № 5482 тип 13

В велокроссе участвуют 276 спортсменов. Специальное устройство регистрирует прохождение каждым из участников промежуточного финиша, записывая его номер с использованием минимально возможного количества бит, одинакового для каждого из участников. Какой объём памяти будет использован устройством, когда промежуточный финиш прошли 240 велосипедистов? (Ответ дайте в байтах.)

Пояснение.

Известно, что с помощью N бит можно закодировать 2^N различных чисел. Поскольку $2^8 < 276 < 2^9$ и для каждого спортсмена число бит одинаково, то для записи каждого из 276 номеров необходимо 9 бит памяти. Поэтому сообщение о 240 номерах имеет объём $240 \cdot 9 = 2160$ бит.

Ваш ответ: *нет ответа*. Правильный ответ: 270

↑ Задание 14 № 14273 тип 14

Исполнитель Редактор получает на вход строку цифр и преобразует её.

Редактор может выполнять две команды, в обеих командах v и w обозначают цепочки цифр.

А) **заменить** (v, w).

Эта команда заменяет в строке первое слева вхождение цепочки v на цепочку w . Например, выполнение команды

заменить (111, 27)

преобразует строку 05111150 в строку 0527150.

Если в строке нет вхождений цепочки v , то выполнение команды **заменить** (v, w) не меняет эту строку.

Б) **нашлось** (v).

Эта команда проверяет, встречается ли цепочка v в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

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).

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 85 идущих подряд цифр 7? В ответе запишите полученную строку.

НАЧАЛО
 ПОКА **нашлось** (333) ИЛИ **нашлось** (777)
 ЕСЛИ **нашлось** (333)
 ТО **заменить** (333, 7)
 ИНАЧЕ **заменить** (777, 3)
 КОНЕЦ ЕСЛИ
 КОНЕЦ ПОКА
 КОНЕЦ

Пояснение.

Данный алгоритм сначала заменит 9 первых семерок на три тройки, а затем заменит эти три тройки обратно на одну семерку. То есть, девять подряд идущих семерок заменяются на одну. Так из 81 семерок получим 9 групп по 9 семерок и еще четыре семерки, всего 13. Снова заменится еще одна группа из 9 семерок, останется 5 семерок. Три первых семерки будут заменены на одну тройку, и останется строка 377.

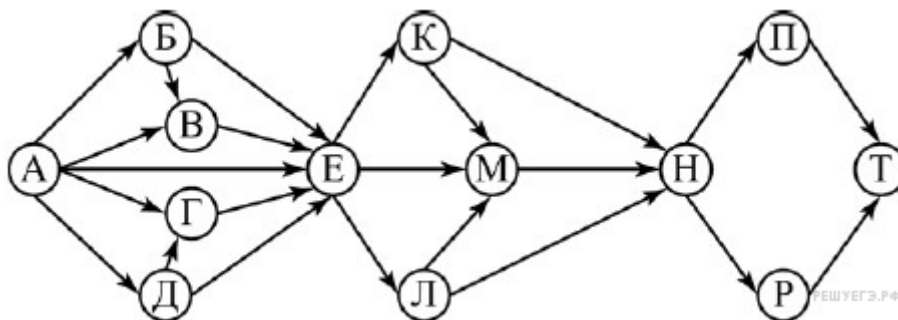
Ответ: 377.

Ваш ответ: *нет ответа*. Правильный ответ: 377

↑ Задание 15 № 14274 тип 15

На рисунке – схема дорог, связывающих города А, Б, В, Г, Д, Е, К, Л, М, Н, П, Р, Т. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.

Сколько существует различных путей из города А в город Т?



Пояснение.

Количество путей до города X равно сумме путей в каждый из тех городов, из которых есть дорога в X. С помощью этого наблюдения посчитаем последовательно количество путей до каждого из городов:

A = 1
 Б = 1
 В = А + Б = 1 + 1 = 2
 Д = 1
 Г = А + Д = 1 + 1 = 2
 Е = А + Б + В + Г + Д = 1 + 1 + 2 + 1 + 2 = 7
 Л = Е = 7
 К = Е = 7

$$\begin{aligned} M &= K + E + L = 7 + 7 + 7 = 21 \\ N &= K + M + L = 7 + 21 + 7 = 35 \\ P &= N = 35 \\ R &= N = 35 \\ T &= P + R = 35 + 35 = 70 \end{aligned}$$

Ответ: 70.

Ваш ответ: *нет ответа*. Правильный ответ: 70

↑ Задание 16 № 2328 тип 16

Запись числа 23_{10} в некоторой системе счисления выглядит так: 212_q . Найдите основание системы счисления q .

Пояснение.

Составим уравнение: $212_q = 2 \cdot q^2 + 1 \cdot q^1 + 2 \cdot q^0 = 23_{10}$, где q - основание этой системы счисления. Решая квадратное уравнение, получаем $q = 3$.

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

↑ Задание 17 № 6899 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) |
|------------------------|--------------------------------|
| Китай & (Япония США) | 390 |
| Китай & Япония | 180 |
| Китай & США & Япония | 50 |

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

Китай & США?

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

Пояснение.

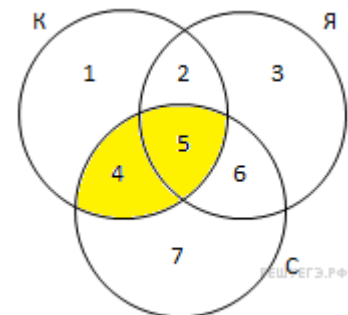
Обозначим количество запросов в данной области как N_i . Наша цель — $N_4 + N_5$. Тогда из таблицы находим, что:

$$\begin{aligned} N_2 + N_5 &= 180, \\ N_5 &= 50, \\ N_4 + N_5 + N_2 &= 390. \end{aligned}$$

Из первого и второго равенств находим: $N_2 = 130$, из последнего равенства: $N_4 + N_5 = 260$.

Ответ: 260.

Ваш ответ: *нет ответа*. Правильный ответ: 260



↑ Задание 18 № 16045 тип 18

Для какого наибольшего целого неотрицательного числа A выражение

$$(y + 2x \neq 48) \vee (A < x) \vee (A < y)$$

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

Пояснение.

Чтобы найти наибольшее целое неотрицательное число A , при котором выражение будет тождественно истинно при любых целых неотрицательных x и y , рассмотрим, в каких случаях условие $(y + 2x \neq 48)$ ложно.

Таким образом, находим все решения, когда $(y = 48 - 2x)$. Это x в промежутке от 0 до 24 и y в промежутке от 48 до 0. Заметим, что для того, чтобы выражение подходило для любых x и y , требуется взять $x = 16$ и $y = 16$. Тогда $A < 16$. Следовательно, наибольшее целое неотрицательное число A будет равняться 15.

Ответ: 15.

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

↑ **Задание 19 № 7703 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 = 0 n = 10 FOR i = 0 TO n-3 s = s+A(i)-A(i+2) NEXT i</pre> | <pre>s = 0 n = 10 for i in range(0,n-2): s = s + A[i] - A[i+2]</pre> |
| Паскаль | Алгоритмический язык |
| <pre>s:=0; n:=10; for i:=0 to n-3 do begin s:=s+A[i]-A[i+2] end;</pre> | <pre>s:=0 n:=10 нц для i от 0 до n-3 s:=s+A[i]-A[i+2] кц</pre> |
| Си++ | |
| <pre>s = 0; n=10; for (i = 0; i <= n-3; i++) s=s+A[i]-A[i+2];</pre> | |

В начале выполнения этого фрагмента в массиве находились двузначные натуральные числа. Какое наибольшее значение может иметь переменная s после выполнения данной программы?

Пояснение.

В переменную s записывается следующая сумма:

$$(A(0) - A(2)) + (A(1) - A(3)) + (A(2) - A(4)) + \dots + (A(6) - A(8)) + (A(7) - A(9)) = A(0) + A(1) - A(8) - A(9).$$

Следовательно, наибольшее значение s равно $99 + 99 - 10 - 10 = 178$.

Ответ: 178.

Ваш ответ: *нет ответа*. Правильный ответ: 178

↑ **Задание 20 № 6781 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 которого алгоритм печатает сначала 3, а потом 2.

| Бейсик | Паскаль |
|---|--|
| <pre> DIM X, A, B AS INTEGER INPUT X A=0: B=0 WHILE X > 0 A = A+1 IF B < (X MOD 8) THEN B = X MOD 8 END IF X = X \ 8 WEND PRINT A PRINT B </pre> | <pre> var x, a, b: integer; begin readln(x); a:=0; b:=0; while x>0 do begin a:=a + 1; if b < (x mod 8) then b:=x mod 8; x:=x div 8; end; writeln(a); write(b); end. </pre> |
| Си++ | Алгоритмический |
| <pre> #include <iostream> using namespace std; int main() { int x, a, b; cin >> x; a=0; b=0; while (x>0){ a = a+1; if (b < (x%8)){ b = x%8; } x = x/8; } cout << a << endl << b << endl; } </pre> | <pre> алг нач цел x, a, b ввод x a:=0; b:=0 нц пока x>0 a:=a+1 если b < mod(x,8) то b:=mod(x,8) все x:=div(x,8) кц вывод a, нс, b кон </pre> |
| Python | |
| <pre> x = int(input()) a = 0 b = 0 while x > 0: a += 1 if b < (x % 8): b = x % 8 x //= 8 print(a) print(b) </pre> | |

Пояснение.

Значение в переменной a после выполнения цикла равно количеству выполненных циклов. Поскольку требуется, чтобы программа напечатала сначала число 3, цикл должен выполниться три раза. Оператор `div` оставляет только целую часть от деления, следовательно, искомое число должно два раза делиться на 8 так, чтобы остаток был не меньше восьми. Следовательно, это число должно быть не меньше числа 64.

В переменную b записывается остаток от деления числа на 8. По условию требуется, чтобы после выполнения цикла переменная b имела значение 2, т. е. остаток от последнего деления на 8 в цикле должен быть равен 2.

Выполним программу для всех чисел, не меньших чем 64. Первое число, которое удовлетворит условию и будет наименьшим. Поскольку программа выводит целые числа и никаких других операторов к числу, кроме операторов div и mod, не применяется, будем рассматривать только целые числа.

При вводе числа 64 программа выведет числа 3 и 1. При вводе числа 65 программа выведет числа 3 и 1. При вводе числа 66 программа выведет числа 3 и 2. Следовательно, ответ 66.

Ответ: 66.

Ваш ответ: *нет ответа*. Правильный ответ: 66

↑ Задание 21 № 6197 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 = -21: B = 21 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 = 12*(9+x)*(9+x)+7 END FUNCTION </pre> | <pre> var a,b,t,M,R :integer; Function F(x: integer):integer; begin F := 12*(9+x)*(9+x)+7; end; BEGIN a := -21; b := 21; 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 12*(9+x)*(9+x)+7; } int main() { int a, b, t, M, R; a = -21; b = 21; 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, R, M a := -21; b := 21 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 := 12*(9+x)*(9+x)+7 кон </pre> |
| Python | |
| <pre> def f(x): return 12*(9+x)*(9+x)+7 a = -21 b = 21 M = a R = f(a) for t in range(a, b+1): if (f(t) > R): M = t R = f(t); </pre> | |

```
print(M)
```

Пояснение.

Алгоритм предназначен для поиска наибольшего t , при котором функция $F(t)$ имеет наибольшее значение на отрезке от a до b . Заметим, что функция является квадратным трёхчленом с положительным старшим коэффициентом, следовательно, наибольшее значение достигается на концах отрезка $[a; b]$. Значение функции в точке 21 больше значения функции в точке -21 , поэтому у переменной M окажется значение 21. Программа выведет на экран число 21.

Ответ: 21.

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

↑ **Задание 22 № 4731 тип 22**

У исполнителя Удвоитель две команды, которым присвоены номера:

1. прибавь 1,
2. умножь на 2.

Первая из них увеличивает на 1 число на экране, вторая удваивает его. Программа для Удвоителя — это последовательность команд. Сколько есть программ, которые число 1 преобразуют в число 20?

Пояснение.

Обозначим $R(n)$ — количество программ, которые преобразуют число 2 в число n . Обозначим $t(n)$ наибольшее кратное 2, не превосходящее n . Заметим, что мы можем получить только числа, кратные 2.

Обе команды увеличивают исходное число, поэтому количество команд не может превосходить $20 - 1 = 19$.

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 $n > 2$).

При $n = 2$ $R(n) = 2$ (два способа: прибавлением единицы и удвоением).

Поэтому достаточно постепенно вычислить значения $R(n)$ для всех чисел, кратных 2 и не превосходящих 20.

Имеем:

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

$$R(4) = 2 + 2 = 4 = R(5),$$

$$R(6) = R(3) + R(4) = 2 + 4 = 6 = R(7),$$

$$R(8) = R(4) + R(6) = 4 + 6 = 10 = R(9),$$

$$R(10) = R(5) + R(8) = 4 + 10 = 14 = R(11),$$

$$R(12) = R(6) + R(10) = 6 + 14 = 20 = R(13),$$

$$R(14) = R(7) + R(12) = 6 + 20 = 26 = R(15),$$

$$R(16) = R(8) + R(14) = 10 + 26 = 36 = R(17),$$

$$R(18) = R(9) + R(16) = 10 + 36 = 46 = R(19),$$

$$R(20) = R(10) + R(18) = 14 + 46 = 60.$$

Ваш ответ: *нет ответа*. Правильный ответ: 60

↑ **Задание 23 № 10424 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:

$$\begin{aligned} (\neg(x_1 \equiv y_1) \rightarrow \neg(x_2 \equiv y_2)) \wedge (x_1 \rightarrow x_2) \wedge (y_1 \rightarrow y_2) &= 1; \\ (\neg(x_2 \equiv y_2) \rightarrow \neg(x_3 \equiv y_3)) \wedge (x_2 \rightarrow x_3) \wedge (y_2 \rightarrow y_3) &= 1; \end{aligned}$$

$$(\neg(x_7 \equiv y_7) \rightarrow \neg(x_8 \equiv y_8)) \wedge (x_7 \rightarrow x_8) \wedge (y_7 \rightarrow y_8) = 1?$$

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.

Будем обозначать набор входных переменных уравнения кортежем из четырёх чисел (x_1, y_1, x_2, y_2) .

Рассмотрим первое уравнение. Переберём все 16 наборов переменных, подойдут 7 из них: $(0, 0, 0, 0), (0, 0, 1, 1), (0, 0, 0, 1), (0, 0, 1, 0), (1, 0, 1, 0), (0, 1, 0, 1), (1, 1, 1, 1)$.

Так как в следующем уравнении встречаются x_2, y_2 , посчитаем, какие пары (x_2, y_2) и сколько раз вошли в наборы, удовлетворяющие первому уравнению: $(0, 0) - 1, (0, 1) - 2, (1, 0) - 2, (1, 1) - 2$.

Будем подставлять эти (x_2, y_2) на место x_3, y_3 во второе уравнение. Так как оно эквивалентно первому уравнению, то можно просто брать x_2, y_2 и находить среди решений первого уравнения кортежи, которые начинаются так же.

То есть $(0, 0)$ даст решения $(0, 0, 0, 0), (0, 0, 1, 1), (0, 0, 0, 1), (0, 0, 1, 0), (0, 1) - (0, 1, 0, 1), (1, 0) - (1, 0, 1, 0), (1, 1) - (1, 1, 1, 1)$.

Посчитаем, сколько каких пар x_3, y_3 получилось среди всех решений системы из первых двух уравнений. Получилось $(0, 0) - 1, (0, 1) - 3, (1, 0) - 3, (1, 1) - 3$.

Перейдя к следующему уравнению, получим $(0, 0) - 1, (0, 1) - 4, (1, 0) - 4, (1, 1) - 4$.

И так далее, с каждым уравнением количество решений будет увеличиваться на 3.

Изначально было 7 решений, плюс ещё 6 уравнений, каждое из которых добавляет по 3 решения.

Итого $7 + 6 \cdot 3 = 25$.

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

Проверка части с развернутым ответом

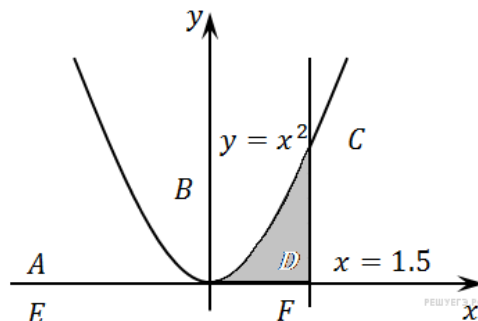
Пожалуйста, оцените решения заданий части с развернутым ответом самостоятельно, руководствуясь указанными критериями.

Задание 24 (С1) № 3643

| Критерии оценивания выполнения задания | Баллы |
|--|-------|
| <p>Обратите внимание! В задаче требуется выполнить три следующих действия.</p> <ol style="list-style-type: none"> 1. Указать для каждой области, как будет работать программа - заполнить таблицу. 2. Исправить ошибку, связанную с неверной расстановкой операторных скобок (отсутствуют случаи ELSE). Исправлением этой ошибки может быть либо добавление случая ELSE к каждому условию IF, либо объединение всех условий IF в одно при помощи конъюнкции. <p>В сложных случаях это действие считается выполненным, если программа выдает одно из двух сообщений «принадлежит» или «не принадлежит» для любых чисел x и y, при этом программа не стала работать хуже, чем раньше, то есть для всех точек, для которых программа ранее выдавала верный ответ, доработанная программа также должна выдавать верный ответ.</p> <ol style="list-style-type: none"> 3. Исправить ошибку в проверяемых условиях. <p>В сложных случаях это действие считается выполненным, если верно определена закрашенная область, то есть программа выводит сообщение «принадлежит» для всех точек закрашенной области и только для них, для точек вне закрашенной области программа выводит «не принадлежит» или не выводит ничего.</p> <p>В целом, критерий оценки работ такой: количество баллов равно количеству выполненных действий. Детально критерии для каждого случая рассмотрены ниже.</p> <p>Задание выполнено полностью:</p> <ol style="list-style-type: none"> 1) Правильно заполнена вся таблица. 2) Написана правильно работающая программа (исправлены обе ошибки), т.е. программа для всех пар чисел x, y верно определяет принадлежность точки закрашенной области. <p>В работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения.</p> | 3 |
| <p>Правильно выполнены два действия из трех.</p> <p>При этом:</p> <ol style="list-style-type: none"> а) Действие по заполнению таблицы засчитывается, даже если ошибки есть, но не более, чем в одной строке. б) При написании операций сравнения допускается одно неправильное использование строгих/нестрогих неравенств (считается несущественной ошибкой, погрешностью записи). Например, вместо «$y \geq -x$» один раз используется «$y > -x$». в) Если ученик написал программу, которая отличается от правильной программы (с учетом п.б) только расстановкой скобок в логических выражениях (например, неправильно учтены приоритеты операций), то считается, что выполнено одно действие по исправлению ошибок из двух. <p>В работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения.</p> | 2 |
| <p>Правильно выполнено только одно действие из трех.</p> <p>При этом:</p> <ol style="list-style-type: none"> а) Действие по заполнению таблицы засчитывается, если ошибки есть не более, чем в двух строках (в отличие от предыдущего пункта). б) При написании операций сравнения допускается любое количество неправильных использований строгих/нестрогих неравенств (т.е. не учитывается корректность работы программы на границах областей) в) Если ученик написал программу, которая отличается от правильной программы только расстановкой скобок в логических выражениях (например, неправильно учтены приоритеты операций), то считается, что выполнено одно действие по исправлению ошибок из двух. <p>В работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения.</p> | 1 |
| <p>Ни один из пунктов задания не выполнен, т.е. не выполнены критерии, позволяющие поставить ученику 1 балл.</p> | 0 |
| <p><i>Максимальный балл</i></p> | 3 |

Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости (x , y — действительные числа) и определяет принадлежность точки заштрихованной области, включая ее границы.

| Паскаль | Бейсик |
|---|--|
| <pre> var x,y: real; begin readln(x,y); if y<=x*x then if x<=1.5 then if y>=0 then write('принадлежит ') else write ('не принадлежит'); end. </pre> | <pre> INPUT x, y IF y<=x*x THEN IF x<=1.5 THEN IF y>=0 THEN PRINT "принадлежит" ELSE PRINT "не принадлежит" ENDIF ENDIF ENDIF END </pre> |
| Си++ | Алгоритмический |
| <pre> int main(void) { float x,y; cin >> x >> y; if (y<=x*x) if (x<=1.5) if (y>=0) cout << "принадлежит"; else cout << "не принадлежит"; } </pre> | <pre> алг нач вещ x,y ввод x,y если y<=x*x то если x<=1.5 то если y>=0 то вывод 'принадлежит' иначе вывод 'не принадлежит' все все кон </pre> |
| Python | |
| <pre> x = float(input()) y = float(input()) if y<=x*x: if x<=1.5: if y>=0: print("принадлежит") else: print("не принадлежит") </pre> | |



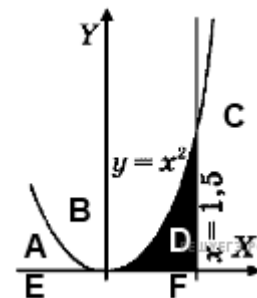
Последовательно выполните следующее:

1. Перерисуйте и заполните таблицу, которая показывает, как работает программа при аргументах, принадлежащих различным областям (A , B , C , D , E и F).

Точки, лежащие на границах областей, отдельно не рассматривать. В столбцах условий укажите "да", если условие выполнится, "нет", если условие не выполнится, "—" (прочерк), если условие не будет проверяться, "не изв.", если программа ведет себя по-разному для разных значений, принадлежащих данной области. В столбце "Программа выведет" укажите, что программа выведет на экран. Если программа ничего не выводит, напишите "—" (прочерк). Если для разных значений,

принадлежащих области, будут выведены разные тексты, напишите "не изв". В последнем столбце укажите "да" или "нет".

2. Укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, достаточно указать любой способ доработки исходной программы.)



| Область | Условие 1 ($y \leq x^2$) | Условие 2 ($x \leq 1.5$) | Условие 3 ($y \geq 0$) | Программа выведет | Область обрабатывается верно |
|---------|-------------------------------|-------------------------------|-----------------------------|-------------------|------------------------------|
| A | | | | | |
| B | | | | | |
| C | | | | | |
| D | | | | | |
| E | | | | | |
| F | | | | | |

Пояснение.

1)

| Область | Условие 1 ($y \leq x^2$) | Условие 2 ($x \leq 1.5$) | Условие 3 ($y \geq 0$) | Программа выведет | Область обрабатывается верно |
|---------|-------------------------------|-------------------------------|-----------------------------|-------------------|------------------------------|
| A | да | да | да | принадлежит | нет |
| B | нет | — | — | — | нет |
| C | да | нет | — | — | нет |
| D | да | да | да | принадлежит | да |
| E | да | да | нет | не принадлежит | да |
| F | да | да | нет | не принадлежит | да |

2) Возможная доработка (Паскаль):

```
if (y <= x*x) and (x <= 1.5) and (y >= 0)
and (x >= 0) then
write('принадлежит ')
else
write('не принадлежит');
```

(могут быть и другие способы доработки).

Задание 25 (С2) № 6349

| 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 MIN (например, присваивается начальное значение, меньшее или равное 1000); 2) при выводе ответа не учитывается, что требуемого числа в массиве может не быть; 3) отсутствует вывод ответа в случае существования минимального числа, удовлетворяющего условию задачи; | 1 |

| | |
|---|---|
| 4) в сравнении с 0 вместо знака «больше» используется знак «больше или равно»; 5) неверно осуществляется проверка делимости на 8; \ 6) на делимость на 8 проверяется не значение элемента, а его индекс; 7) в сложном условии вместо логической операции «И» используется логическая операция «ИЛИ»; 8) используется переменная, не объявленная в разделе описания переменных; 9) не указано или неверно указано условие завершения цикла; 10) индексная переменная в цикле не меняется (например, в цикле while) или меняется неверно; 11) неверно расставлены операторные скобки | |
| Ошибок, перечисленных в п. 1–11, две или больше, или алгоритм сформулирован неверно | 0 |
| Максимальный балл | 2 |

Дан целочисленный массив из 30 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 произведение всех таких элементов массива, которые являются двузначными числами с чётной суммой цифр. Гарантируется, что в исходном массиве есть хотя бы один элемент, значение которого является двузначным числом, и при этом сумма его цифр чётна. 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. Исходные данные всегда подобраны так, что результат произведения не выходит за пределы объявленных типов данных.

| Бейсик | Паскаль |
|--|--|
| <pre>N = 30 DIM A(N) AS LONG DIM I, J, P AS LONG FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre> | <pre>const N = 30; var a: array [1..N] of longint; i, j, p: longint; begin for i := 1 to N do readln(a[i]); ... end.</pre> |
| Си++ | Алгоритмический язык |
| <pre>#include <iostream> using namespace std; #define N 30 int main() { long a[N]; long i, j, p; for (i = 0; i < N; i++) cin >> a[i]; ... }</pre> | <pre>алг нач цел N = 30 целтаб a[1:N] цел i, j, p нц для i от 1 до N ввод a[i] кц ... кон</pre> |
| Естественный язык | |
| Объявляем массив A из 30 элементов. Объявляем целочисленные переменные I, J, P. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й. ... | |
| Python | |
| <pre># допускается также # использовать две # целочисленные переменные j, p a = [] n = 30 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.

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла).

| Бейсик | Паскаль |
|--|--|
| <pre>P = 1 FOR I = 1 TO N IF A(I) >= 10 AND A(I) <= 99 AND (A(I) MOD 10 + A(I) DIV 10) MOD 2 = 0 THEN P = P * A(I) END IF NEXT I PRINT P</pre> | <pre>p := 1; for i := 1 to N do if (a[i] >= 10) and (a[i] <= 99) and ((a[i] mod 10 + a[i] div 10) mod 2 = 0) then p := p * a[i]; writeln(p);</pre> |
| Си | Алгоритмический язык |
| <pre>p = 1; for (i = 0; i < N; i++) if (a[i] >= 10 && a[i] <= 99 && (a[i] % 10 + a[i] / 10) % 2 == 0) p *= a[i]; cout << p;</pre> | <pre>p := 1 нц для i от 1 до N если a[i] >= 10 и a[i] <= 99 и mod(mod(a[i], 10) + div(a[i], 10), 2) = 0 то p := p * a[i] все кц вывод p</pre> |
| Естественный язык | |
| <p>Записываем в переменную P начальное значение, равное 1. В цикле от первого элемента до тридцатого находим целую часть и остаток от деления элемента исходного массива на 10. Теперь находим остаток от деления суммы полученных двух значений на 2. Если результат равен 0 и значение текущего элемента массива больше 9 и меньше 100, то считаем произведение данного элемента массива и значения переменной P. Результат умножения сохраняем в переменную P. Переходим к следующему элементу массива. После завершения цикла выводим значение переменной P.</p> | |

Задание 26 (С3) № 16053

| Критерии оценивания выполнения задания | Баллы |
|--|-------|
| Все три задания выполнены верно | 3 |
| Не выполнены условия, позволяющие поставить 3 балла, при этом верно выполнены любые два задания | 2 |
| Не выполнены условия, позволяющие поставить 2 балла, при этом верно выполнено одно любое задание | 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 лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) **один** камень или увеличить количество камней в куче в **три раза**. Например,

пусть в одной куче 10 камней, а в другой 7 камней; такую позицию в игре будем обозначать $(10, 7)$. Тогда за один ход можно получить любую из четырёх позиций: $(11, 7)$, $(30, 7)$, $(10, 8)$, $(10, 21)$. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 68. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, при которой в кучах будет 68 или больше камней.

В начальный момент в первой куче было шесть камней, во второй куче — S камней; $1 \leq S \leq 61$.

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 играющего по этой стратегии игрока, не являющиеся для него безусловно выигрышными, т.е. не являющиеся выигрышными независимо от игры противника.

Выполните следующие задания.

Задание 1.

а) Укажите все такие значения числа S , при которых Петя может выиграть за один ход.

б) Известно, что Ваня выиграл своим первым ходом после неудачного первого хода Пети. Укажите минимальное значение S , когда такая ситуация возможна.

Задание 2. Укажите такое значение S , при котором у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

— Петя не может выиграть за один ход;

— Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Для указанного значения S опишите выигрышную стратегию Пети.

Задание 3. Укажите значение S , при котором одновременно выполняются два условия:

— у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;

— 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).

В узлах дерева указывайте позиции, на рёбрах рекомендуется указывать ходы. Дерево не должно содержать партии, невозможные при реализации выигрывающим игроком своей выигрышной стратегии. Например, полное дерево игры не является верным ответом на это задание.

Пояснение.

Задание 1.

а) Петя может выиграть при $21 \leq S \leq 61$.

б) $S = 7$.

Задание 2.

Возможное значение S : 20. В этом случае Петя, очевидно, не может выиграть первым ходом. Однако он может получить позицию $(7, 20)$. После хода Вани может возникнуть одна из четырёх позиций: $(8, 20)$, $(21, 20)$, $(7, 21)$, $(7, 60)$. В каждой из этих позиций Петя может выиграть одним ходом, утроив количество камней во второй куче.

Замечание для проверяющего. Ещё одно возможное значение S для этого задания – число 13. В этом случае Петя первым ходом должен утроить количество камней в меньшей куче и получить позицию $(6 * 3, 13) = (18, 13)$.

При такой позиции Ваня не может выиграть первым ходом, а после любого хода Вани Петя может выиграть, утроив количество камней в большей куче.

Достаточно указать одно значение S и описать для него выигрышную стратегию.

Задание 3

Возможное значение S : 19. После первого хода Пети возможны позиции: $(7, 19)$, $(18, 19)$, $(6, 20)$, $(6, 57)$. В позициях $(18, 19)$ и $(6, 57)$ Ваня может выиграть первым ходом, утроив количество камней во второй куче. Из позиций $(7, 19)$ и $(6, 20)$ Ваня может получить позицию $(7, 20)$. Эта позиция разобрана в п. 2. Игрок, который её получил (теперь это Ваня), выигрывает своим вторым ходом.

В таблице изображено дерево возможных партий (и только их) при описанной стратегии Вани. Заключительные позиции (в них выигрывает Ваня) выделены жирным шрифтом. 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) |
|--------------------|---|--|---|--|
| (6, 19) | $(6, 19+1) = \mathbf{(6, 20)}$ | $(6+1, 20) = \mathbf{(7, 20)}$ | $(7+1, 20) = \mathbf{(8, 20)}$ | $(8, 20*3) = \mathbf{(8, 60)}$ |

| | | | | | |
|------------------|---|---|--|---|---|
| Всего: 25 | Всего: 26 | Всего: 27 | Всего: 28 | Всего: 68 | |
| | | | $(7, 20+1)=(7, 21)$ Всего: 28 | $(7, 21*3)=(7, 63)$ Всего: 70 | |
| | | | $(7*3, 20) = (21, 20)$ Всего: 41 | $(21, 20*3)=(21, 60)$ Всего: 81 | |
| | | | | $(7, 20*3)=(7, 60)$ Всего: 67 | $(7, 60*3)=(7, 180)$ Всего: 187 |
| | $(6+1, 19)=(7, 19)$ Всего: 26 | $(7, 19+1)=(7, 20)$ Всего: 27 | $(7+1, 20)=(8, 20)$ Всего: 28 | $(8, 20*3)=(8, 60)$ Всего: 68 | |
| | | | $(7, 20+1)=(7, 21)$ Всего: 28 | $(7, 21*3)=(7, 63)$ Всего: 70 | |
| | | | $(7*3, 20) = (21, 20)$ Всего: 41 | $(21, 20*3)=(21, 60)$ Всего: 81 | |
| | | | $(7, 20*3)=(7, 60)$ Всего: 67 | $(7, 60*3)=(7, 180)$ Всего: 187 | |
| | | | $(6*3, 19) = (18, 19)$ Всего: 37 | $(18, 19*3) = (18, 57)$ Всего: 75 | |
| | | | $(6, 19*3)=(6, 57)$ Всего: 63 | $(6, 57*3) = (6, 171)$ Всего: 177 | |

Примечание для эксперта. Дерево всех партий может быть также изображено в виде ориентированного графа — так, как показано на рисунке, или другим способом. Важно, чтобы множество полных путей в графе находилось во взаимно однозначном соответствии со множеством партий, возможных при описанной в решении стратегии.

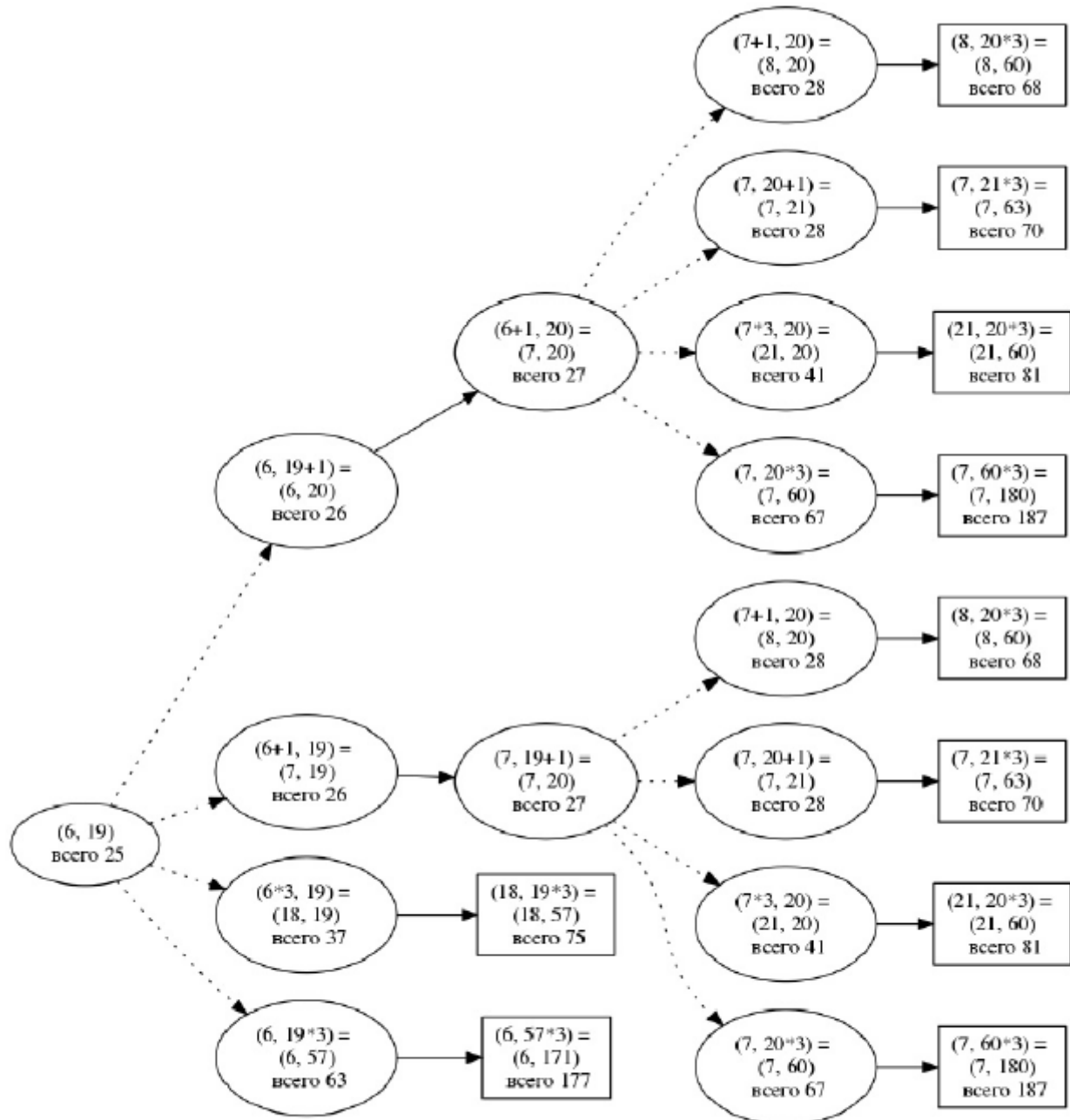


Рис. 1. Дерево всех партий, возможных при Ваниной стратегии. Ходы Пети показаны пунктиром; ходы Вани – сплошными линиями. Прямоугольником обозначены позиции, в которых партия заканчивается.

Замечание для проверяющего. Не является ошибкой указание только одного заключительного хода выигрывающего игрока в ситуации, когда у него есть более одного выигрышного хода.

Задание 27 (С4) № 14242

| Критерии оценивания выполнения задания | Баллы |
|--|-------|
| <p><i>Пояснения для проверяющих.</i></p> <p>1. Задание Б является усложнением задания А. Если в качестве решения задания Б представлено решение задания А, то согласно приведённым ниже критериям его оценка будет такой же, как если бы это решение было представлено в качестве решения задания А.</p> <p>2. Два задания (и, соответственно, возможность для экзаменуемого представить две программы) дают ученику возможность (при его желании) сначала написать менее сложное и менее эффективное решение (задание А), которое даёт ему право получить 2 балла, а затем приступить к поиску более эффективного решения.</p> <p>3. Приведённые в п. 2.1–2.5 правила имеют целью избежать снижения оценки из-за того, что ученик перепутал обозначения заданий</p> | |
| Критерии оценивания задания А | |

| | |
|---|--------------|
| При решении задачи А программа верно находит требуемую сумму для любых 6 пар исходных данных. Допускается до пяти синтаксических и приравненных к ним ошибок (см. критерии оценивания задания Б на 4 балла) | 2 |
| Не выполнены условия, позволяющие поставить 2 балла. Из описания алгоритма и общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи. Допускается любое количество «описок» | 1 |
| Не выполнены критерии, позволяющие поставить 1 или 2 балла | 0 |
| <i>Максимальный балл для задания А</i> | 2 |
| Критерии оценивания выполнения задания Б | Баллы |
| <p>Программа правильно работает для любых соответствующих условию входных данных и при этом эффективна как по времени, так и по памяти, т.е. не используются массивы и другие структуры данных (в том числе стек рекурсивных вызовов), размер которых зависит от количества входных элементов, а время работы пропорционально этому количеству. Возможно использование массивов и динамических структур данных при условии, что в них в каждый момент времени хранится фиксированное количество элементов, требующих для хранения меньше 1Кб.</p> <p>Программа может содержать не более трёх синтаксических ошибок 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>К синтаксическим ошибкам приравнивается использование неверного типа данных. Если одна и та же ошибка встречается несколько раз, она считается за одну ошибку</p> | 4 |
| <p>Невыполнены условия, позволяющие поставить 4 балла. Программа в целом работает правильно для любых входных данных произвольного размера. Время работы пропорционально количеству введённых чисел, правильно указано, какие величины должны вычисляться по ходу чтения 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 или отсутствие инициализации счётчиков; 2) использование нахождения остатка вместо деления (mod вместо div в Паскале) или наоборот; 3) ошибка при построении цикла разбиения на цифры (если в программе используется такой цикл); 4) отсутствие инициализации или неверная инициализация максимума; 5) допущен выход за границу массива; 6) вместо максимума ищется минимум; 7) при наличии нескольких ответов выводится не максимальный из них; 8) вместо суммы (индекса массива счётчиков) выводится количество (значение элемента) | 3 |
| <p>Не выполнены условия, позволяющие поставить 3 или 4 балла.</p> <p>Программа работает в целом верно, эффективно или нет, например для решения задачи используется перебор всех возможных вариантов выбора элементов в парах. В реализации алгоритма допускается до трёх содержательных ошибок, допустимые виды ошибок перечислены в критериях на 3 балла.</p> <p>Количество синтаксических «описок» не должно быть более семи. Программа может быть неэффективна по времени, например все числа запоминаются в массиве и перебираются все возможные суммы, т.е., по сути, реализовано решение задачи А без ограничений на количество введённых пар</p> | 2 |
| <p>Не выполнены условия, позволяющие поставить 2, 3 или 4 балла. Из описания алгоритма или общей структуры программы видно, что экзаменуемый в целом правильно представляет путь решения задачи независимо от эффективности. При этом программа может быть представлена отдельными фрагментами, без ограничений на количество синтаксических и содержательных ошибок. 1 балл ставится также за решения, верные лишь в частных случаях</p> | 1 |
| Не выполнены критерии, позволяющие поставить 1, 2, 3 или 4 балла | 0 |
| <i>Максимальный балл для задания Б</i> | 4 |
| <i>Итоговый максимальный балл</i> | 4 |

Дан набор из N целых положительных чисел. Для каждого числа вычисляется сумма двух последних цифр в его десятичной записи (для однозначных чисел предпоследняя цифра считается равной нулю). Необходимо определить, какая сумма при этом получается чаще всего. Если таких сумм несколько, необходимо вывести **наибольшую** из них.

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 1 Кбайт 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 10000.

Пример входных данных:

```
5
15
417
123
6
4841
```

Пример выходных данных для приведённого выше примера входных данных:

```
6
Суммы двух последних цифр для чисел из данного набора равны 6, 8, 5, 6, 5.
Чаще других (по два раза) встречаются 6 и 5, в ответе выводится бóльшая из этих сумм.
```

Пояснение.

Сумма двух цифр может принимать значения от 0 до 18. Необходимо создать массив из 19 элементов с индексами от 0 до 18 и использовать его для подсчёта количества чисел с соответствующими суммами двух последних цифр. Использование массива не делает программу неэффективной по памяти, так как размер массива не зависит от N . Затем нужно найти значение максимального элемента этого массива и вывести максимальный из индексов элементов, равных этому максимуму.

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

```
var
  N: integer; {количество чисел}
  a: integer; {очередное число}
  s: integer; {сумма двух последних цифр}
  d: array [0..18] of integer; {подсчёт сумм}
  mx: integer; {максимальное количество сумм}
  imx: integer; {самая частая сумма}
  i: integer;
begin
  for i:=0 to 18 do d[i]:=0;
  readln(N);
  for i:=1 to N do begin
    readln(a);
    s := a mod 10 + a div 10 mod 10;
    d[s] := d[s]+1;
  end;
  mx := 0;
  for i:=0 to 18 do begin
    if d[i] >= mx then begin
```



```
mx := d[i];
imx := i;

end;
writeln(imx)
end.
```

Возможно также «лобовое» решение: запишем все исходные числа в массив, найдём суммы двух последних цифр этих чисел и отсортируем их в порядке убывания, после чего найдём сумму, которая встречается чаще всего. Такое решение не является эффективным ни по памяти (требуемая память зависит от размера исходных данных), ни по времени (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}
  val: integer; {самая частая сумма}
  a: array [1..1000] of integer;
  max_lenght: integer;
  i, j, k, lenght: integer;
begin
  readln(N);
  for i:=1 to N do read(a[i]);
  for i:=1 to N do a[i] := a[i] mod 10 + a[i] div 10 mod 10;
  for i:=1 to N-1 do
    for j:=1 to N-i do
      if a[j] < a[j+1] then begin
        k := a[j];
        a[j] := a[j+1];
        a[j+1] := k;
      end;
  max_lenght := 0;
  lenght := 0;
  val := a[1];
  for i := 1 to N do
    if a[i]=a[i+1] then
      lenght := lenght + 1
    else if lenght>max_lenght then begin
      max_lenght := lenght;
      val := a[i];
      lenght := 0;
    end;
  writeln(val);
end.
```