

Решения

↑ Задание 1 № 13558 тип 1

Вычислите: $10101101_2 - 255_8 + D_{16}$. Ответ запишите в десятичной системе счисления. В ответе запишите только число, основание системы счисления писать не нужно.

Пояснение.

Переведём все числа в десятичную систему счисления и вычислим.

$$10101101_2 = 173_{10}; 255_8 = 173_{10}; D_{16} = 13_{10}.$$

$$173_{10} - 173_{10} + 13_{10} = 13_{10}.$$

Ответ: 13.

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

↑ Задание 2 № 10278 тип 2

Логическая функция F задаётся выражением:

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

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

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

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

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

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

Тогда 1-му столбцу соответствует переменная y , а 2-му столбцу соответствует переменная x . В ответе нужно написать: yx .

Пояснение.

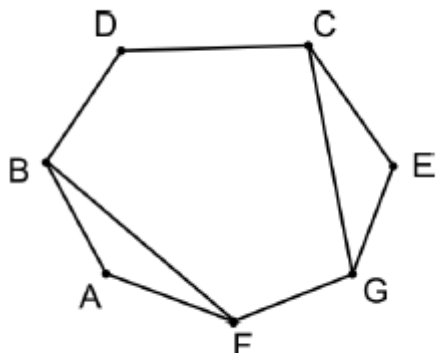
Выражение равно 1, если хотя бы одна из двух скобок равна 1. Первая скобка равна 1 при наборах переменных $(0, 0, 1)$ и $(0, 1, 1)$. Вторая скобка только при $(0, 0, 0)$. Из третьего набора выводов не сделать, из первых же двух понятно, что переменные идут в порядке x, y, z (x оба раза 0, в первом столбце оба раза 0; z оба раза 1, третий столбец оба раза тоже 1).

Ответ: xyz .

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

↑ Задание 3 № 16030 тип 3

На рисунке слева изображена схема дорог Н-ского района, в таблице звёздочкой обозначено наличие дороги из одного населённого пункта в другой. Отсутствие звёздочки означает, что такой дороги нет.



| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | | * | * | |
| 2 | | | * | * | | | * |
| 3 | | * | | | * | | * |
| 4 | | * | | | | * | |
| 5 | * | | * | | | * | |
| 6 | * | | | * | * | | |
| 7 | | * | * | | | | |

Каждому населённому пункту на схеме соответствует его номер в таблице, но неизвестно, какой именно номер. Определите, какие номера населённых пунктов в таблице могут соответствовать населённым пунктам В и С на схеме. В ответе запишите эти два номера в возрастающем порядке без пробелов и знаков препинания.

Пояснение.

Заметим, что D - единственная вершина степени 2, которая связана с вершинами степени 3 — В и С, связанными с остальными вершинами степени 2. Значит, D соответствует П4. Таким образом, населённым пунктам В и С соответствуют пункты П2 и П6.

Ответ: 26.

Ваш ответ: *нет ответа*. Правильный ответ: 26

↑ Задание 4 № 3762 тип 4

Сколько записей удовлетворяют условию «Пол = 'м' и Графы > Кодирование»?

| Фамилия | Пол | Адресация | Графы | Закономерности | Кодирование | Алгоритмы |
|-----------|-----|-----------|-------|----------------|-------------|-----------|
| Боровец | м | 66 | 64 | 64 | 66 | 62 |
| Грибников | м | 82 | 64 | 67 | 64 | 82 |
| Дарецкая | ж | 61 | 67 | 66 | 69 | 61 |
| Житников | м | 69 | 68 | 86 | 66 | 64 |
| Манникова | ж | 66 | 65 | 66 | 61 | 66 |
| Соловкина | ж | 68 | 60 | 81 | 72 | 76 |

Пояснение.

Условию "Пол = 'м'" удовлетворяют "Боровец, Грибников, Житников".

Условию "Графы > Кодирование" удовлетворяют "Житников, Манникова".

Одновременно к двум критериям относится только один человек "Житников".

Ответ: 1.

Ваш ответ: *нет ответа*. Правильный ответ: 1

↑ Задание 5 № 15790 тип 5

По каналу связи передаются сообщения, содержащие только семь букв: А, Б, Г, И, М, Р, Я. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: А — 010, Б — 011, Г — 100. Какое **наименьшее** количество двоичных знаков потребуется для кодирования слова МАГИЯ?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

Пояснение.

Следующую буква должна кодироваться как 11, поскольку 10 мы взять не можем. 100 взять не можем из-за Г, значит следующая буква должна быть закодирована кодом 101. Следующая буква должна кодироваться как 000, поскольку 00 взять не можем, иначе не останется кодовых слов для оставшейся буквы, которые удовлетворяют условию Фано. Значит, последняя буква будет кодироваться как 001. Тогда наименьшее количество двоичных знаков, которые потребуются для кодирования слова МАГИЯ равно $2 + 3 + 3 + 3 + 3 = 14$.

Ответ: 14.

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

↑ Задание 6 № 16882 тип 6

Автомат обрабатывает натуральное число N ($0 \leq N \leq 255$) по следующему алгоритму:

1. Строится восьмьбитная двоичная запись числа N .
2. Все цифры двоичной записи заменяются на противоположные (0 на 1, 1 на 0).
3. Полученное число переводится в десятичную запись.
4. Из нового числа вычитается исходное, полученная разность выводится на экран.

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

1. Восьмьбитная двоичная запись числа N : 00001101.
2. Все цифры заменяются на противоположные, новая запись 11110010.
3. Десятичное значение полученного числа 242.
4. На экран выводится число $242 - 13 = 229$.

Какое число нужно ввести в автомат, чтобы в результате получилось 111?

Пояснение.

Заметим, что инверсия двоичной восьмьбитной записи числа в сумме с исходным числом дает 11111111, то есть 255. (В исходном примере: $00001101 + 11110010 = 11111111$.) Следовательно, если исходное число равно N , то инвертированное число равно $255 - N$. Затем автомат осуществляет вычитание, вычисляя $255 - 2N$.

Поэтому, чтобы найти число, которое нужно ввести в автомат для получения 111, нужно решить уравнение $255 - 2N = 111$. Тем самым, искомое число равно 72.

Ответ: 72.

Ваш ответ: *нет ответа*. Правильный ответ: 72

↑ Задание 7 № 7360 тип 7

В электронной таблице значение формулы =СРЗНАЧ(В5:Е5) равно 80. Чему равно значение формулы =СУММ(В5:Д5), если значение ячейки Е5 равно 20? Пустых ячеек в таблице нет.

Пояснение.

Среднее значение величин, хранящихся в ячейках В5, С5, Д5, Е5 равно 80, значит, сумма этих значений равна $4 \cdot 80 = 320$. Учитывая, что значение в ячейке Е5 равно 20, находим, что сумма значений в ячейках В5, С5, Д5 равна $320 - 20 = 300$.

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

↑ Задание 8 № 6924 тип 8

Определите, что будет напечатано в результате выполнения программы (записанной ниже на разных языках программирования):

| Бейсик | Паскаль |
|--|---|
| <pre> DIM N, S AS INTEGER N = 0 S = 0 WHILE S <= 365 S = S + 12 N = N + 5 WEND PRINT N </pre> | <pre> var n, s: integer; begin n := 0; s := 0; while s <= 365 do begin s := s + 12; n := n + 5 end; write(n) end. </pre> |
| Си++ | Алгоритмический язык |
| <pre> #include <iostream> using namespace std; int main() { int n, s; n = 0; s = 0; while (s <= 365) { s = s + 12; n = n + 5; } cout << n << endl; } </pre> | <pre> алг нач цел n, s n := 0 s := 0 нц пока s <= 365 s := s + 12 n := n + 5 кц вывод n кон </pre> |
| Python | |
| <pre> n = 0 s = 0 while s <= 365: s += 12 n += 5 print(n) </pre> | |

Пояснение.

Цикл while выполняется до тех пор, пока истинно условие $s \leq 365$, т. е. переменная s определяет, сколько раз выполнится цикл.

Заметим, что $\left\lfloor \frac{365}{12} \right\rfloor = 30$. На 31 шаге s станет равной 372 и условие $s \leq 365$ окажется невыполненным, цикл прервется. Следовательно, значение n будет равно $5 \cdot 31 = 155$.

Ответ: 155.

Ваш ответ: *нет ответа*. Правильный ответ: 155

↑ Задание 9 № 2418 тип 9

У Васи есть доступ к Интернет по высокоскоростному одностороннему радиоканалу, обеспечивающему скорость получения им информации 2^{17} 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^{16} бит в секунду. Петя договорился с Васей, что тот будет скачивать для него данные объемом 8 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 8 Мбайт по каналу со скоростью передачи данных 2^{16} бит/с; к этому времени нужно добавить задержку файла у Васи (пока он не получит 1024 Кбайта данных по каналу со скоростью 2^{17} бит/с).

Переведём объём информации из мегабайт в биты: $Q = 8 \text{ Мбайт} = 8 * 2^{20} \text{ байт} = 2^{26} \text{ бит}$.

Время задержки: $t_0 = 1024 \text{ кб} / 2^{17} \text{ бит/с} = 2^{(10 + 10 + 3) - 17} \text{ с} = 2^6 \text{ с} = 64 \text{ с}$.

Время скчивания данных Петей: $t_1 = 2^{26} \text{ бит} / 2^{16} \text{ бит/с} = 2^{10} \text{ с} = 1024 \text{ с}$.

Полное время: $t = t_0 + t_1 = (1024 + 64) \text{ с} = 1088 \text{ с}$.

Ответ: 1088.

Ваш ответ: *нет ответа*. Правильный ответ: 1088

↑ **Задание 10 № 15850 тип 10**

Все 4-буквенные слова, составленные из букв А, И, О, У, Э, записаны в алфавитном порядке и пронумерованы.

Вот начало списка:

1. АААА
2. АААИ
3. АААО
4. АААУ

...

Под каким номером стоит ИААЭ?

Пояснение.

Заменяем буквы А, И, О, У, Э на 0, 1, 2, 3, 4 соответственно (для них порядок очевиден — по возрастанию).

Выпишем начало списка, заменив буквы на цифры:

1. 0000
2. 0001
3. 0002
4. 0003

...

Полученная запись есть числа, записанные в пятеричной системе счисления в порядке возрастания. Слово ИААЭ можно представить в виде: $1004_5 = 129_{10}$. Так как порядковый номер на единицу больше, получаем ответ: 130.

Ответ: 130.

Ваш ответ: *нет ответа*. Правильный ответ: 130

↑ **Задание 11 № 7372 тип 11**

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

$F(n) = 1$ при $n = 1$;

$F(n) = F(n - 1) \cdot n$ при $n \geq 2$.

Чему равно значение функции $F(6)$?
В ответе запишите только натуральное число.

Пояснение.

Последовательно найдём значения F :

$F(1) = 1,$
 $F(2) = 2,$
 $F(3) = 6,$
 $F(4) = 24,$
 $F(5) = 120.$
 $F(6) = 720.$

Таким образом, ответ $F(6) = 720$.

Ответ: 720.

Ваш ответ: *нет ответа*. Правильный ответ: 720

↑ Задание 12 № 3541 тип 12

В терминологии сетей TCP/IP маской подсети называется 32-разрядное двоичное число, определяющее, какие именно разряды IP-адреса компьютера являются общими для всей подсети – в этих разрядах маски стоит 1. Обычно маски записываются в виде четверки десятичных чисел – по тем же правилам, что и IP-адреса. Для некоторой подсети используется маска 255.255.248.0. Сколько различных адресов компьютеров допускает эта маска?

Примечание. На практике для адресации компьютеров не используются два адреса: адрес сети и широковещательный адрес.

Пояснение.

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

2. Запишем число 248 в двоичном виде: $248 = 11111000_2$. В конце этого числа стоят 3 нуля, еще 8 нулей мы получаем из последнего октета маски. Итого есть 11 двоичных разрядов для того, чтобы записать адрес компьютера.

3. Далее, $2^{11} = 2048$, так как два адреса не используются, получаем $2048 - 2 = 2046$.

Ваш ответ: *нет ответа*. Правильный ответ: 2046

↑ Задание 13 № 7195 тип 13

Для регистрации на сайте некоторой страны пользователю требуется придумать пароль. Длина пароля — ровно 7 символов. В качестве символов используются десятичные цифры и 30 различных букв местного алфавита, причём все буквы используются в двух начертаниях: как строчные, так и прописные (регистр буквы имеет значение!). Под хранение каждого такого пароля на компьютере отводится минимально возможное и одинаковое целое количество байтов, при этом используется посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством битов. Определите объём памяти, который занимает хранение 40 паролей. (Ответ дайте в байтах.)

Пояснение.

Всего для кодирования может быть использовано 10 десятичных цифр, 30 строчных и 30 прописных букв, т. е. 70 символов. Известно, что с помощью N бит можно закодировать 2^N различных символов. Поскольку $2^6 < 70 < 2^7$ и для каждого пароля число бит одинаково, то для записи каждого из 7 символов необходимо 7 бит памяти. Для хранения всех 7 символов номера нужно 49 бит, а т. к. для записи используется целое число байт, то необходимо округлить в большую сторону число 49 до числа, кратного восьми, это число $56 = 7 \cdot 8$ бит (7 байт). Поэтому для хранения 40 паролей понадобится $40 \cdot 7 = 280$ байт.

Ваш ответ: *нет ответа*. Правильный ответ: 280

↑ **Задание 14 № 1815 тип 14**

Исполнитель КОРАБЛИК «живет» в ограниченном прямоугольном водоеме-лабиринте, разделенном на клетки и изображенном на рисунке (вид сверху). Серые клетки — скалистые берега, светлые — свободное пространство, безопасное для передвижения КОРАБЛИКА. По краю водоема-лабиринта также находятся скалы с нанесенными на них номерами и буквами для удобства идентификации клеток.

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 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 КОРАБЛИК:

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

При выполнении любой из этих команд КОРАБЛИК 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 КОРАБЛИК (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 КОРАБЛИК разбивается о скалы.

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

НАЧАЛО

ПОКА <справа свободно> вправо

ПОКА <слева свободно> влево

вверх

влево

КОНЕЦ

Пояснение.

Разбиться Кораблик может только при выполнении команд "вверх" и "влево". Начав из любой клетки строк 1, 3, 4, 5, 7, столбца А и клеток Е6–J6 кораблик разобьётся, выполняя команду влево. Стартовав из клеток С2, Е2, G2, I2, K2, M2, O2, O4, O6, O8 кораблик уцелеет.

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

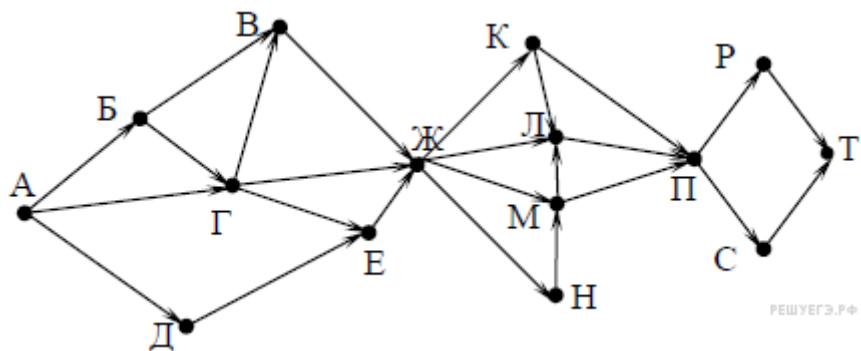
↑ **Задание 15 № 15137 тип 15**

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

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 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 | |

направлении, указанном стрелкой.

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



Пояснение.

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

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

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

$$A = 1.$$

$$B = A = 1.$$

$$Г = A + B = 2.$$

$$D = A = 1.$$

$$B = B + Г = 3.$$

$$E = Г + D = 3.$$

$$Ж = B = 3. \text{ (E и Г не учитываем, поскольку в этих вершинах не проходим через B).}$$

$$K = Ж = 3.$$

$$H = Ж = 3.$$

$$M = Ж + H = 6.$$

$$L = K + Ж + M = 12.$$

$$П = K + L + M = 21.$$

$$P = П = 21.$$

$$C = П = 21.$$

$$T = P + C = 42.$$

Примечание. Необходимо найти количество различных путей из города А в город Т, проходящих через город В.

Ответ: 42.

Ваш ответ: *нет ответа*. Правильный ответ: 42

↑ Задание 16 № 6991 тип 16

В системе счисления с основанием N запись числа 79_{10} оканчивается на 2, а запись числа 111_{10} — на 1. Чему равно число N?

Пояснение.

Так как запись чисел оканчивается на 1 и 2, то основание системы счисления не может быть меньше трёх. Последняя цифра в записи числа — это остаток от деления числа на основание системы счисления. Подбором находим, что условию удовлетворяет только $N = 11$.

Ответ: 11.

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

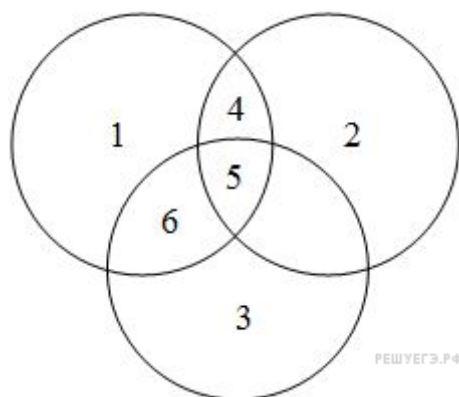
↑ **Задание 17 № 13493 тип 17**

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

| Запрос | Найдено страниц (в тысячах) |
|--------------------------------|--------------------------------|
| Мадрид & Севилья | 290 |
| Мадрид & Барселона | 355 |
| Мадрид & (Барселона Севилья) | 465 |

Какое количество страниц (в тысячах) будет найдено по запросу Мадрид & Барселона & Севилья? Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов

Пояснение.



Построим диаграмму Венна для данной задачи.

Количество запросов в данной области будем обозначать N_i . Наша цель — найти N_5

При этом круг 1 соответствует Мадрид, круг 2 — Барселоне, круг 3 — Севилье.

Из таблицы находим, что:

$$N_5 + N_6 = 290$$

$$N_5 + N_4 = 355$$

$$N_6 + N_4 + N_5 = 465$$

Тогда находим:

$$N_4 = 175$$

$$N_6 = 110$$

$$N_5 = 180$$

Ответ: 180.

Ваш ответ: *нет ответа*. Правильный ответ: 180

↑ **Задание 18 № 7763 тип 18**

На числовой прямой даны два отрезка: $P = [5, 30]$ и $Q = [14, 23]$. Укажите наибольшую возможную длину промежутка A , для которого формула

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

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

Пояснение.

Знаком \sim обозначается операция эквивалентности (результат $X \sim Y$ — истина, если значения X и Y совпадают).

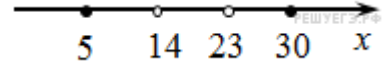
Введем обозначения:

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

Тогда, применив преобразование импликации, получаем:

$$\neg(P \sim Q) \vee \neg A \Leftrightarrow \neg(P \sim Q) \vee \neg A = 1.$$

Выражение $\neg(P \sim Q)$ истинно только тогда, когда $x \in [5; 14)$ и $x \in (23; 30]$ (см. рисунок). В таком случае, для того, чтобы выражение было истинно при любом x , A должно лежать либо в промежутке $[5; 14)$, либо $(23; 30]$. Следовательно, наибольшая возможная длина промежутка равна $14 - 5 = 9$.



Ответ: 9.

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

↑ Задание 19 № 10322 тип 19

Ниже представлен записанный на разных языках программирования фрагмент одной и той же программы. В программе описан одномерный целочисленный массив A , в представленном фрагменте программы обрабатываются элементы массива с индексами от 1 до 10.

| Бейсик | Python |
|---|--|
| <pre>s = 0 n = 10 FOR i = 2 TO n s = s+A(i)*A(i)-A(i-1)*A(i-1) NEXT i</pre> | <pre>s=0 n=10 for i in range(1,n): s=s + A[i]*A[i]-A[i-1]*A[i-1]</pre> |
| Паскаль | Алгоритмический язык |
| <pre>s:=0; n:=10; for i:= 2 to n do begin s:=s+A[i]*A[i]-A[i-1]*A[i-1]; end;</pre> | <pre>s:=0 n:=10 нц для i от 2 до n s:=s + A[i]*A[i]-A[i-1]*A[i-1] кц</pre> |
| Си++ | |
| <pre>s = 0; n=10; for (i = 1; i < n; i++) { s=s + A[i]*A[i]-A[i-1]*A[i-1]; }</pre> | |

В начале выполнения этого фрагмента в массиве находились числа 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, то есть $A[1] = 1$, $A[2] = 2$ и так далее. Чему будет равно значение переменной s после выполнения данной программы?

Пояснение.

$$s = 2^2 - 1^2 + 3^2 - 2^2 + 4^2 - 3^2 + \dots + 10^2 - 9^2 = -1^2 + (2^2 - 2^2) + (3^2 - 3^2) + \dots + (9^2 - 9^2) + 10^2 = 100 - 1 = 99$$

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

↑ Задание 20 № 3263 тип 20

Ниже записана программа. Получив на вход число x , эта программа печатает два числа, L и M . Укажите наибольшее из таких чисел x , при вводе которых алгоритм печатает сначала 3, а потом 7.

| Бейсик | Python |
|---|--|
| <pre> DIM X, L, M AS INTEGER INPUT X L = 0 M = 0 WHILE X > 0 L = L + 1 IF M < x THEN M = X MOD 10 ENDIF X = X \ 10 WEND PRINT L PRINT M </pre> | <pre> x = int(input()) L = 0 M = 0 while x > 0: L += 1 if M < x: M = x % 10 x = x // 10 print(L) print(M) </pre> |
| Паскаль | Алгоритмический язык |
| <pre> var x, L, M: integer; begin readln(x); L := 0; M := 0; while x > 0 do begin L := L + 1; if M < x then M := x mod 10; x := x div 10; end; writeln(L); writeln(M); end. </pre> | <pre> алг нач цел x, L, M ввод x L := 0 M := 0 нц пока x > 0 L := L + 1 если M < x то M := mod(x, 10) все x := div(x, 10) кц вывод L, M кон </pre> |
| Си++ | |
| <pre> #include <iostream> using namespace std; int main() { int x, L, M; cin >> x; L = 0; M = 0; while (x > 0){ L = L + 1; if(M < x){ M = x % 10; } x = x / 10; } cout << L << endl << M endl; } </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 только целую часть от деления, то при делении на 10 это равносильно отсечению последней цифры.

Из приведенного цикла видно, что на каждом шаге от десятичной записи x отсекается последняя цифра до тех пор, пока все цифры не будут отсечены, то есть x не станет равно 0; поэтому цикл выполняется столько раз, сколько цифр в десятичной записи введенного числа, при этом число L столько же раз увеличивается на 1. Следовательно, конечное значение L совпадает с числом цифр в x . Для того, чтобы L стало $L = 3$, x должно быть **трёхзначным**.

Теперь рассмотрим оператор изменения M :

```
if M < x then begin
M:=x mod 10;
end;
```

Оператор `mod` оставляет только остаток от деления, при делении на 10 это последняя цифра.

После первого шага M может быть любым однозначным числом, т. к. на втором шаге число x двузначное, а однозначное число всегда меньше любого двузначного, а значит, M может приобрести интересующее нас значение 7. Поэтому третья цифра числа x $x(3) = 9$.

Вторая цифра уже не может быть любой, так как при $x(2) > x(1)$ значение переменной M уже не поменяется на третьем шаге. Поэтому $x(2)$ не превышает 7. Поскольку мы хотим получить наибольшее число x , мы берём $x(1) = 7$. Осталось верно определить $x(2)$.

Число x достигнет максимума при $x(2)=7$, при этом на третьем шаге переменная M уже не изменится, потому что нарушено условие $M < x$ ($7 = 7$).

Окончательно получаем: $x = 779$.

Ответ: 779.

Ваш ответ: *нет ответа*. Правильный ответ: 779

↑ Задание 21 № 13470 тип 21

Напишите в ответе наименьшее значение входной переменной k , при котором программа выдаёт ответ 15. Для Вашего удобства программа приведена на пяти языках программирования.

| Бейсик | Python |
|---|--|
| <pre> DIM K, I AS LONG INPUT K I = 1 WHILE F(I) < G(I)+K I = I + 1 WEND PRINT I FUNCTION F(N) F = (N+1) * (N+1) END FUNCTION FUNCTION G(N) G = N * N END FUNCTION </pre> | <pre> def f(n): return (n+1)*(n+1) def g(n): return n*n i = 1 k = int(input()) while f(i) < g(i) + k: i+=1 print (i) </pre> |
| Паскаль | Алгоритмический язык |
| <pre> var k, i : longint; function f(n: longint) : longint; begin f := (n+1) * (n+1); end; function g(n: longint) : longint; begin g:= n * n; </pre> | <pre> алг нач цел i, k ввод k i := 1 нц пока f(i) < g(i) + k i := i + 1 кц вывод i все </pre> |

| | |
|---|--|
| <pre> end; begin readln(k); i := 1; while f(i) < g(i) + k i := i+1; writeln(i) end. </pre> | <pre> кон алг цел f(цел n) нач знач := (n+1) * (n+1) кон алг цел g(цел n) нач знач := n * n кон </pre> |
| Си++ | |
| <pre> #include <iostream> using namespace std; long f(long n) { return (n+1) * (n+1); } long g(long n) { return n * n; } int main() { long k, i; cin >> k; i = 1; while (f(i) < g(i)+ k) { i++; } cout << i << endl; } </pre> | |

Пояснение.

Запишем уравнение: $x^2 + 2x + 1 < x^2 + k$. Алгоритм должен выполняться ровно 14 раз, следовательно, нужное значение для x достигается при $k - 1 > 2 \cdot 14$, но при этом $k - 1 < 2 \cdot 15$, тогда получаем, что $k > 29$, однако, $k < 31$, значит, $k = 30$.

Ответ: 30.

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

↑ Задание 22 № 13579 тип 22

Исполнитель Осень16 преобразует число на экране.

У исполнителя есть три команды, которым присвоены номера:

- 1) **Прибавить 1;**
- 2) **Прибавить 2;**
- 3) **Прибавить 3.**

Первая команда увеличивает число на экране на 1, вторая увеличивает его на 2, третья — увеличивает на 3.

Программа для исполнителя Осень16 — это последовательность команд.

Сколько существует программ, для которых при исходном числе 1 результатом является число 15 и при этом траектория вычислений содержит число 8?

Траектория вычислений программы — это последовательность результатов выполнения всех команд программы. Например, для программы 121 при исходном числе 7 траектория будет состоять из чисел 8, 10, 11.

Пояснение.

Искомое количество программ равно произведению количества программ, получающих из числа 1 число 8, на количество программ, получающих из числа 8 число 15.

Пусть $R(n)$ — количество программ, которые число 3 преобразуют в число n , а $P(n)$ — количество программ, которые число 8 преобразуют в число n .

Для всех $n > 3$ верны следующие соотношения:

1. $R(n) = R(n - 1) + R(n - 2) + R(n - 3)$, так как существует три способа получения n — прибавлением единицы, прибавлением двойки или прибавлением тройки. Аналогично $P(n) = P(n - 1) +$

$$P(n - 2) + P(n - 3)$$

Последовательно вычислим значения $R(n)$:

$$R(1) = 1$$

$$R(2) = 1$$

$$R(3) = 2$$

$$R(4) = R(1) + R(2) + R(3) = 4$$

$$R(5) = R(4) + R(3) + R(2) = 4 + 2 + 1 = 7$$

$$R(6) = R(5) + R(4) + R(3) = 7 + 4 + 2 = 13$$

$$R(7) = R(6) + R(5) + R(4) = 13 + 7 + 4 = 24$$

$$R(8) = R(7) + R(6) + R(5) = 24 + 13 + 7 = 44$$

Теперь вычислим значения $P(n)$:

$$P(8) = 1$$

$$P(9) = 1$$

$$P(10) = 2$$

$$P(11) = 4$$

$$P(12) = P(11) + P(10) + P(9) = 4 + 2 + 1 = 7$$

$$P(13) = P(12) + P(11) + P(10) = 7 + 4 + 2 = 13$$

$$P(14) = P(13) + P(12) + P(11) = 13 + 7 + 4 = 24$$

$$P(15) = P(14) + P(13) + P(12) = 24 + 13 + 7 = 44$$

Таким образом, количество программ, удовлетворяющих условию задачи равно $44 \cdot 44 = 1936$.

Ответ: 1936.

Ваш ответ: *нет ответа*. Правильный ответ: 1936

↑ Задание 23 № 5819 тип 23

Сколько существует различных наборов значений логических переменных x_1, x_2, \dots, x_{11} , которые удовлетворяют всем перечисленным ниже условиям?

$$\neg(x_1 \equiv x_2) \wedge ((x_1 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3)) = 0$$

$$\neg(x_2 \equiv x_3) \wedge ((x_2 \wedge \neg x_4) \vee (\neg x_2 \wedge x_4)) = 0$$

...

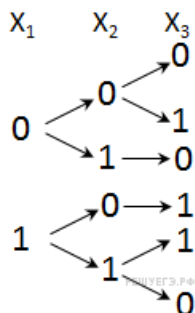
$$\neg(x_9 \equiv x_{10}) \wedge ((x_9 \wedge \neg x_{11}) \vee (\neg x_9 \wedge x_{11})) = 0$$

В ответе **не нужно** перечислять все различные наборы значений переменных x_1, x_2, \dots, x_{11} при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

Пояснение.

Рассмотрим первое уравнение.

При $x_1 = 1$ возможны два случая: $x_2 = 0$ и $x_2 = 1$. В первом случае $x_3 = 1$. Во втором — x_3 либо 0, либо 1. При $x_1 = 0$ также возможны два случая: $x_2 = 0$ и $x_2 = 1$. В первом случае x_3 либо 0, либо 1. Во втором — $x_3 = 0$. Таким образом, уравнение имеет 6 решений (см. рисунок).

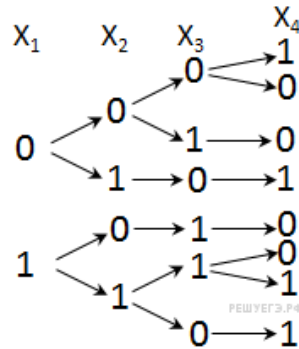


Рассмотрим систему из двух уравнений.

Пусть $x_1 = 1$. При $x_2 = 0$ возможен лишь один случай: $x_3 = 1$, переменная $x_4 = 0$. При $x_2 = 1$ возможно два случая: $x_3 = 0$ и $x_3 = 1$. В первом случае $x_4 = 1$, во втором — x_4 либо 0, либо 1. Всего имеем 4 варианта.

Пусть $x_1 = 0$. При $x_2 = 1$ возможен лишь один случай: $x_3 = 0$, переменная $x_4 = 1$. При $x_2 = 0$ возможно два случая: $x_3 = 0$ и $x_3 = 1$. В первом случае x_4 либо 1, либо 0, во втором — $x_4 = 0$. Всего имеем 4 варианта.

Таким образом, система из двух уравнений имеет $4 + 4 = 8$ вариантов (см. рисунок).



Система из трёх уравнений будет иметь 10 решений, из четырёх — 12. Система из девяти уравнений будет иметь 22 решения.

Ответ: 22.

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

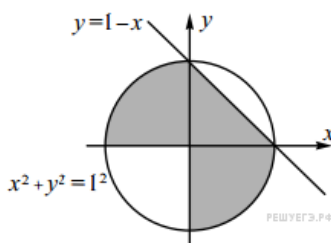
Проверка части с развернутым ответом

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

Задание 24 (С1) № 6469

| Критерии оценивания выполнения задания | Баллы |
|--|-------|
| Выполнены все три действия | 3 |
| Правильно выполнены два действия из трёх. При этом 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 работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения | 2 |
| Правильно выполнено только одно действие из трёх, то есть либо только приведена таблица, которая содержит ошибки в не более чем одной строке, либо таблица не приведена (или приведена и содержит ошибки более чем в одной строке), но исправлена одна ошибка программы. При оценивании этого задания на 1 балл допускается не учитывать корректность работы программ на точках границ областей (вместо нестрогих неравенств в решении были использованы строгие неравенства или наоборот). В работе (во фрагментах программ) допускается наличие отдельных синтаксических ошибок, не искажающих замысла автора решения | 1 |
| Ни один из пунктов задания не выполнен, то есть не выполнены критерии, позволяющие поставить ученику 1 балл | 0 |
| Максимальный балл | 3 |

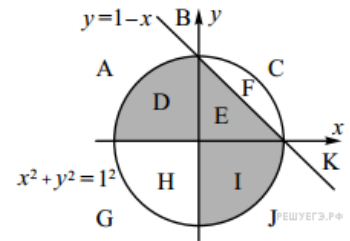
Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости (x, y — действительные числа) и определяет принадлежность точки заштрихованной области (включая границы). Программист торопился и написал программу неправильно. Ниже для Вашего удобства программа представлена на четырёх языках программирования.



| Бейсик | Паскаль |
|--|--|
| <pre> INPUT x, y IF y<=1-x THEN IF y>=0 THEN IF x>=0 THEN IF x*x+y*y<=1 THEN PRINT "принадлежит" ELSE PRINT "не принадлежит" END IF END IF END IF END IF END </pre> | <pre> var x,y: real; begin readln(x,y); if y<=1-x then if y>=0 then if x>=0 then if x*x+y*y<=1 then write('принадлежит') else write('не принадлежит'); end. </pre> |
| Си++ | Алгоритмический язык |
| <pre> #include <iostream> using namespace std; int main() { float x,y; cin >> x >> y; if (y<=1-x) if (y>=0) if (x>=0) if (x*x+y*y<=1) cout << "принадлежит"; else cout << "не принадлежит"; } </pre> | <pre> алг нач вещ x,y ввод x,y если y<=1-x то если y>=0 то если x>=0 то если x*x+y*y<=1 то вывод 'принадлежит' иначе вывод 'не принадлежит' все все все все кон </pre> |
| Python | |
| <pre> x,y = float(input()) if y<=1-x: if y>=0: if x>=0: if x*x+y*y<=1: 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, F, G, H, I, J, K). Точки, лежащие на границах областей, отдельно не рассматривать. Координатные оси не являются границами областей. В столбцах условий укажите «да», если условие выполнится, «нет», если условие не выполнится, «—» (прочерк), если условие не будет проверяться, «не изв.» (если программа ведёт 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».



| Область | Условие1 ($y \leq 1-x$) | Условие 2 ($y >= 0$) | Условие 3 ($x >= 0$) | Условие 4 ($x^2+y^2 \leq 1$) | Программа выведет | Область обрабатывается верно |
|---------|---------------------------|------------------------|------------------------|--------------------------------|-------------------|------------------------------|
| A | | | | | | |
| B | | | | | | |
| C | | | | | | |
| D | | | | | | |
| E | | | | | | |
| F | | | | | | |
| G | | | | | | |
| H | | | | | | |
| I | | | | | | |
| J | | | | | | |
| K | | | | | | |

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

Пояснение.

Элементы ответа:

| Область | Условие1 ($y \leq 1-x$) | Условие 2 ($y >= 0$) | Условие 3 ($x >= 0$) | Условие 4 ($x^2+y^2 \leq 1$) | Программа выведет | Область обрабатывается верно |
|---------|---------------------------|------------------------|------------------------|--------------------------------|-------------------|------------------------------|
| A | да | да | нет | — | — | нет |
| B | нет | — | — | — | — | нет |
| C | нет | — | — | — | — | нет |
| D | да | да | нет | — | — | нет |
| E | да | да | да | да | принадлежит | да |
| F | нет | — | — | — | — | нет |
| G | да | нет | — | — | — | нет |
| H | да | нет | — | — | — | нет |
| I | да | нет | — | — | — | нет |
| J | да | нет | — | — | — | нет |
| K | нет | — | — | — | — | нет |

Возможная доработка (Паскаль, объединение двух пересекающихся областей, с отсечением):

```
if (x*y<=1) and (y<=1-x) and ((x>=0) or (y>=0))
then
write('принадлежит')
else write('не принадлежит')
```

Возможная доработка (Си, отдельно рассматриваем три области, лежащие по разные стороны от координатных осей):

```
if ((x*y<=1) && (y>=0) && (x<=0) ||
(x*y<=1) && (y<=0) && (x>=0) ||
(x>=0) && (y>=0) && (y<=1-x))
cout << "принадлежит";
else
cout << "не принадлежит";
```

Обратите внимание! При разделении области вдоль какой-либо линии точки, которые лежат на этой линии внутри области, могут быть причислены к одной части, к другой или к обеим (то есть, например, в приведенном решении на языке Паскаль одно из условий $x >= 0$ или $x <= 0$) может быть строгим) Могут быть и другие верные способы доработки.

Задание 25 (C2) № 2921

| 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 перемен- | 2 |

| | |
|---|---|
| ных 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. | |
| В любом варианте решения может присутствовать не более одной ошибки из числа следующих: 1) не инициализируется или неверно инициализируется переменная; 2) при выводе ответа не учитывается, что требуемого числа в массиве может не быть; 3) отсутствует вывод ответа; 4) в сравнении перепутан знак; 5) неверно осуществляется проверка делимости; 6) на делимость 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 | 1 |
| Ошибок, перечисленных в п. 1–11, две или больше, или алгоритм сформулирован неверно | 0 |
| <i>Максимальный балл</i> | 2 |

Дан целочисленный массив из 30 элементов. Элементы массива могут принимать значения от 0 до 1000. Опишите на русском языке или на одном из языков программирования алгоритм, который позволяет подсчитать и вывести среднее арифметическое элементов массива, имеющих нечетное значение. Гарантируется, что в исходном массиве хотя бы один элемент имеет нечетное значение.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

| Бэйсик | Паскаль |
|--|--|
| <pre>N=30 DIM A(N) AS INTEGER DIM I, X, Y AS INTEGER DIM S AS SINGLE FOR I = 1 TO N INPUT A(I) NEXT I ... END</pre> | <pre>const N=30; var a: array [1..N] of integer; i, x, y: integer; s: real; begin for i:=1 to N do readln(a[i]); ... end.</pre> |
| Си++ | Естественный язык |
| <pre>#include <iostream> using namespace std; #define N 30 int main(void) {int a[N]; int i, x, y; float s; for (i=0; i<N; i++) cin >> a[i]; ... }</pre> | <p>Объявляем массив A из 30 элементов. Объявляем целочисленные переменные I, X, Y. Объявляем вещественную переменную S. В цикле от 1 до 30 вводим элементы массива A с 1-го по 30-й. ...</p> |
| Python | |
| <pre># допускается также # использовать две # целочисленные переменные x, y # и вещественную переменную s 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, Borland Pascal 7.0) или в виде блок-схемы. В этом случае вы должны использовать переменные, аналогичные переменным, используемым в алгоритме, записанном на естественном языке, с учетом синтаксиса и особенностей используемого вами языка программирования.

Пояснение.

Содержание верного ответа
(допускаются иные формулировки ответа, не искажающие его смысла)
На языке Паскаль

```
x:=0;
y:=0;
for i:=1 to N do
if (a[i] mod 2=1) then begin
x:=x+a[i];
y:=y+1;
end;
s:=x/y;
writeln(s);
```

```

На языке Бейсик
X = 0
Y = 0
FOR I = 1 TO N
IF A(I) MOD 2 = 1 THEN
X = X + A(I)
Y = Y + 1
ENDIF
NEXT I
S = X / Y
PRINT S

```

На языке Си

```

x=0;
y=0;
for (i=0; i<N; i++)
if (a[i]%2==1)
{ x=x+a[i];
y++;
}
s=(float)x/y;
cout << s << endl;

```

На естественном языке

Записываем в переменные X и Y начальное значение, равное нулю. В цикле от первого элемента до тридцатого находим остаток от деления элемента исходного массива на два. Если этот остаток равен единице, то увеличиваем счетчик суммы X на значение текущего элемента массива, а счетчик количества Y на 1. Переходим к следующему элементу. После цикла производим деление счетчика суммы X на счетчик количества Y и записываем результат в переменную S.

Выводим значение переменной S.

Задание 26 (С3) № 4570

| Критерии оценивания выполнения задания | Баллы |
|---|-------|
| Правильное указание выигрывающего игрока и его ходов со строгим доказательством правильности (с помощью или без помощи дерева игры) | 3 |
| При наличии в представленном решении одного из пунктов. 1. Правильное указание выигрывающего игрока, стратегии игры, приводящей к победе, но при отсутствии доказательства её правильности. 2. Правильно указан выигрывающий игрок, приведено дерево игры, но отсутствует обоснование правильности выигрывающей стратегии. 3. Правильно указан выигрыш Пети, правильно указан первый ход, рассмотрены все варианты ответа Вани, для каждого из них правильно указан выигрывающий ответ Пети. Однако анализ игры не доведён до конца, и отсутствует обоснование стратегии | 2 |
| При наличии в представленном решении одного из пунктов. 1. Победитель указан неправильно, однако указаны все варианты первого хода Пети, и для каждого из них указан (выигрывающий, по мнению экзаменуемого) ответ Вани, но в анализе дальнейших действий были допущены ошибки. 2. Правильно указан выигрыш Пети, правильно указан первый ход, но описание выигрышной стратегии неполно, и для некоторых (больше одного, но не всех) вариантов ответа Вани правильно указан выигрывающий ответ Пети | 1 |
| Задание не выполнено, или в представленном решении полностью отсутствует описание элементов выигрышной стратегии и отсутствует анализ вариантов первого и второго ходов играющих (даже при наличии правильного указания выигрывающего игрока) | 0 |
| Максимальный балл | 3 |

Два игрока, Петя и Ваня, играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых – 4, а во второй – 3 камня. У каждого игрока неограниченно много камней. Игроки ходят по очереди, первый ход делает Петя. Ход состоит в том, что игрок или утраивает число камней в какой-либо куче, или добавляет 2 камня в какую-либо кучу. Игра завершается в тот момент, когда количество камней в одной из куч становится не менее 19. Если в момент завершения игры общее число камней в двух кучах не менее 35, то выиграл Ваня, в противном случае – Петя. Кто выигрывает при безошибочной игре обоих игроков? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

Пояснение.

Выигрывает Петя, своим первым ходом он должен добавить 2 камня к первой куче. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны пары чисел, разделённые запятой. Эти числа соответствуют количеству камней на каждом этапе игры в первой и второй кучах соответственно.

| Позиция после первого хода (Петя) | 2-й ход | 3-й ход | 4-й ход | 5-й ход |
|-----------------------------------|--------------------------|------------------------|---|--|
| 6, 3 | Ваня (все варианты хода) | Петя (выигрышные ходы) | Ваня (все варианты хода, кроме непосредственно проигрышных) | Петя (выигрышные ходы, экзаменуемому достаточно указать один из вариантов) |
| | 6, 5 | 6, 7 | 18, 7 | 20, 7 |
| | | | 8, 7 | 24, 7 ИЛИ 8, 21 |
| | | | 6, 9 | 6, 27 |
| | 18, 3 | 20, 3 | Выигрыш Пети | |
| 6, 9 | 6, 27 | | | |
| | 8, 3 | 24, 3 | | |

Таблица содержит все возможные варианты ходов второго игрока. Из неё видно, что при любом ответе второго игрока у первого имеется ход, приводящий к победе.

Задание 27 (С4) № 6590

| Критерии оценивания выполнения задания | Баллы |
|---|-------|
| Программа работает для любых входных данных произвольного размера и находит ответ, не сохраняя входные данные в массиве, размер которого соответствует числу N (количество запросов). Программа просматривает входные данные один раз, сохраняя в массиве размером 12 данные о количестве запросов, поданных на каждую из встретившихся в списке задач (и учитывает, что в списке их может быть и меньше 12). Допускается наличие в тексте программы трёх синтаксических ошибок: пропущен или неверно указан знак пунктуации; неверно написано или пропущено зарезервированное слово языка программирования; не описана или неверно описана переменная; применяется операция, недопустимая для соответствующего типа данных (если одна и та же ошибка встречается несколько раз, то это считается за одну ошибку) | 4 |
| Программа работает верно, но входные данные запоминаются в массиве, размер которого соответствует числу N. Этот массив, возможно, потом сортируется. Допускается наличие до пяти синтаксических ошибок, описанных выше. Возможно, в принципе верно организованном вводе данных есть одна ошибка (например, использование read вместо readln в Паскале или неверное считывание строки в C++). 3 балла также выставляется, если в эффективной программе, удовлетворяющей критериям выставления 4 баллов, есть одна ошибка, в результате которой программа работает неверно на некоторых наборах нетипичных входных данных (например, все запросы относятся к одной и той же задаче) | 3 |
| Программа работает в целом верно, эффективно или нет, но в реализации алгоритма содержится до двух ошибок (неверная инициализация счётчиков, хотя в предложенных выше решениях обнулять их не требуется. Возможно, программа работает неверно, если в списке упомянуто меньше 12 задач, выход за границу массива, допущена ошибка в принципиально верно организованной сортировке или алгоритме поиска минимальных элементов, используется знак "<" вместо "<=", "or" вместо "and" и т.п.). Возможно, некорректно организовано считывание входных данных. Допускается наличие до семи синтаксических ошибок, описанных выше | 2 |
| Программа, возможно, работает неверно при некоторых входных данных, но по приведённому тексту решения ясно, что экзаменуемый понимает, из каких этапов должно состоять решение задачи. При использовании сортировки она может быть реализована принципиально неверно (например, вместо двух циклов используется один), или допущена принципиальная ошибка в поиске трёх наименее популярных элементов. Всего допускается до четырёх различных ошибок в реализации алгоритма, в том числе описанных в критериях присвоения 2 баллов. Допускается наличие любого количества синтаксических ошибок, описанных выше | 1 |
| Задание не выполнено или выполнено неверно | 0 |
| <i>Максимальный балл</i> | 4 |

В телевизионном танцевальном марафоне с определением победителя с помощью телезрителей после каждого тура объявляется sms-голосование, в котором зрители указывают наиболее понравившуюся им пару из максимум 10 пар, которые участвуют в проекте. Вам предлагается написать эффективную, в том числе по используемой памяти, программу, которая будет обрабатывать результаты sms-голосования по данному вопросу. Результаты голосования получены в виде номеров пар (каждый элемент списка соответствует одному sms-сообщению). Следует учитывать, что количество голосов в списке может быть очень велико. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи. На вход программе в первой строке подаётся количество пришедших sms-сообщений N. В каждой из последующих N строк записан номер пары от 1 до 10. *Пример входных данных:*

```
4
2
10
3
2
```

Программа должна вывести список всех пар, встречающихся в списке, в порядке возрастания (неубывания) количества голосов, отданных за ту или иную пару, с указанием количества отданных за неё голосов. При этом каждая пара должна быть выведена ровно один раз вне зависимости от того, сколько раз она встречается в списке. *Пример выходных данных для приведённого выше примера входных данных:*

```
3 1
10 1
2 2
```

Пояснение.

Программа читает все входные данные один раз, не запоминая все входные данные в массиве, размер которого равен N, а сразу подсчитывая в массиве пар, сколько голосов отдано за каждую из них. После окончания ввода производится одновременная сортировка массивов номеров пар и количества голосов, отданных за них, в порядке возрастания количества голосов; затем выводится список пар, за которые подано ненулевое число голосов, с указанием частоты встречаемости. Баллы начисляются только за программу, которая решает задачу хотя бы для одного частного случая. Ниже приведены примеры решения задания на языках Паскаль и Бейсик. Допускаются решения, записанные на других языках программирования. При оценивании решений на других языках программирования необходимо учитывать особенности этих языков программирования.

| Пример правильной и эффективной программы на языке Паскаль: | Пример правильной и эффективной программы на языке Бейсик: |
|--|--|
| <pre>Var n, i, j, t: integer; Count, Names: array[1..10] of integer; Begin For i := 1 to 10 do begin Count[i] := 0; Names[i] := i; end; ReadLn(N); {Считываем количество голосов} for i:=1 to N do begin ReadLn(t); {считали очередную пару} Count[t] := Count[t] + 1;{учитываем поданный голос} end; {Сортируем массивы Names и Count в порядке возрастания значений массива Count} for i:=10 downto 2 do for j:=2 to i do if Count[j-1]>Count[j] then begin t:=Count[j]; Count[j]:=Count[j-1]; Count[j-1]:=t;</pre> | <pre>DIM Count(10) AS INTEGER DIM Names(10) AS INTEGER FOR i = 1 TO 10 Count(i) = 0 Names(i) = i NEXT i INPUT N 'Считываем количество голосов FOR i = 1 TO N INPUT t 'считали пару 'Учитываем поданный голос Count(t) = Count(t) + 1 NEXT i 'Сортируем массивы Names и Count в порядке возрастания 'значений массива Count FOR i = 10 TO 2 STEP -1 FOR j = 2 TO i IF Count(j - 1) > Count(j) THEN t = Count(j): Count(j) = Count(j - 1): Count(j - 1) = t t = Names(j): Names(j) = Names(j - 1): Names(j - 1) = t</pre> |

```
t:=Names[j]; Names[j]:=Names[j-1]; Names[j-1]:=t;
end;
for i:=1 to 10 do
if Count[i] > 0 then
writeln(Names[i], ' ', Count[i]);
end.
```

```
END IF
NEXT j
NEXT i
FOR i = 1 TO 10
IF Count(i) > 0 THEN PRINT Names(i), Count(i)
NEXT i
END
```