

**Открытая олимпиада школьников (информатика)
(№64 Перечня олимпиад школьников, 2020/2021 уч.год)**

Содержание

Содержание	1
Задания для 11 класса	1
Заключительный этап	1
Отборочный этап. Первый тур	10
Отборочный этап. Второй тур	14
Задания для 9 и 10 класса	19
Заключительный этап	19
Отборочный этап. Первый тур	27
Отборочный этап. Второй тур	30
Задания для 7 и 8 класса	34
Заключительный этап	34
Отборочный этап. Первый тур	47
Отборочный этап. Второй тур	50

Задания для 11 класса

Заключительный этап (приведен один из вариантов заданий)

1. Кодирование информации. Системы счисления (2 балла)

[Периодическая система]

Дана система уравнений, неизвестными в которой являются основания двух позиционных систем счисления X и Y :

$$\begin{cases} 0, (34)_X + 0, (13)_Y = 1 \\ 2 * 101_X - 102_Y = 1 \end{cases}$$

где левая часть первого уравнения представляет собой сумму двух периодических дробей.

Найдите и запишите в ответ через пробел сначала значение X , а затем значение Y , удовлетворяющие этой системе уравнений. Если таких пар несколько, запишите пару с наименьшим значением X . Если такой пары не существует, запишите в ответ NULL.

Ответ: 5 7

Решение:

Определим область допустимых значений: $x \geq 5; y \geq 4$

Перепишем первое уравнение, используя развернутую запись чисел в системах счисления с основаниями X и Y :

$$\frac{3x + 4}{x^2 - 1} - \frac{y + 3}{y^2 - 1} = 1$$

Обратим внимание на знаменатели.

Перепишем второе уравнение, используя развернутую запись чисел в системах счисления с основаниями X и Y :

$$2(x^2 + 1) - (y^2 + 2) = 1$$

Преобразуем уравнение так, чтобы была возможность заменой удобно привести первое уравнение к общему знаменателю:

$$\begin{aligned} 2x^2 + 2 - y^2 - 2 &= 1 \\ 2x^2 - 1 &= y^2 \\ 2(x^2 - 1) &= y^2 - 1 \end{aligned}$$

Подставляем в первое уравнение, приводим к общему знаменателю:

$$(3x + 4) \cdot 2 + \sqrt{2x^2 - 1} + 3 = 2 \cdot (x^2 - 1)$$

Решим уравнение, понимая, что нас интересует только целый положительный корень, удовлетворяющий ОДЗ. Он единственный: $x = 5$. Соответственно, находим $y = 7$ и получаем ответ «5 7».

2. Кодирование информации. Объем информации (2 балла)

[ABC]

Петя решил построить генератор случайных последовательностей символов. Генератор состоит из трех источников символов, общего буфера и таймера. Таймер отсчитывает секунды, начиная с 1. Источник 1 каждую секунду записывает в буфер один символ, по кругу выбирая символы из набора $\{A, B, C\}$. Источник 2 каждую секунду, кратную 2, записывает в буфер один символ A. Источник 3 каждую секунду, кратную 3, записывает в буфер один символ, по кругу выбирая символы из набора $\{A, B\}$. Если в некоторую секунду несколько источников должны записать в буфер свои символы, они всегда делают это по порядку возрастания их номеров.

Вот результаты работы генератора на первых 6 секундах:

1. A
2. ABA
3. ABACA
4. ABACAAA
5. ABACAAAB
6. ABACAAABCAB

Петя настолько уверен в надежности своего генератора, что, сформировав в буфере строку из N символов, решил записать её в память как последовательность двоичных кодов отдельных символов, используя для кода каждого символа одинаковое минимально возможное количество бит. Вася проанализировал работу генератора Пети и убедился, что он не такой уж надежный. Вася увидел, что длина строки Пети N кратна трем, и разбил строку в буфере на тройки идущих подряд символов, начиная с начала строки. Он посчитал, сколько всего различных троек символов встречается в строке, и решил записать в память строку в виде последовательности двоичных кодов троек символов, используя для кода каждой тройки одинаковое, минимально возможное количество бит. Вася обнаружил, что ему потребовалось для записи в память строки на 40 бит меньше, чем Пете. Определите и запишите в ответ длину строки N.

Ответ:60

Решение:

Проанализируем построение строк на первых 6 секундах работы генератора и построим строку, которая получится на 7-ой секунде. В скобках указано, какие источники и какие символы добавляют на соответствующем шаге.

1. A (Источник 1, A)
2. ABA (Источник 1, B; Источник 2, A)
3. ABACA (Источник 1, C; Источник 3, A)
4. ABASAAA (Источник 1, A; Источник 2, A)
5. ABASAAAB (Источник 1, B)
6. ABASAAABCSAB (Источник 1, C; Источник 2, A; Источник 3, B)
7. ABASAAABCSABA (Источник 1, A)

Обратим внимание, что на секунде 7 повторилось состояние всех источников: источник 1 добавляет символ A, источник 2 готов на следующей (четной) секунде добавить символ A и источник 3 готов через секунду (на секунде 9) добавить символ A. То есть, начинается такой же цикл из 6 секунд, который прибавит в конец строки, полученной на 6-ой секунде её копию еще через 6 секунд.

Обратим внимание, что длина строки на 6-ой секунде была 11 символов. То, есть, если мы повторим цикл из 6 секунд 3 раза, на 18-ой секунде получится строка из трех одинаковых идущих друг за другом подстрок, длиной 33 символа (кратно трем). А затем все будет повторяться. Следовательно, все возможные тройки символов содержатся в этой строке из 33 символов. Получим её, склеив три копии строки, полученной на 6 секунде, и выделим тройки:

ABACAAABCSABABASAAABCSABABASAAABCSAB

Обратим внимание, что тройка ABA повторяется два раза, следовательно, всего в этом фрагменте строки, а следовательно и в любой строке большей длины (по условию кратной трем), могут встретиться только 10 различных троек.

Теперь посчитаем затраты памяти. В строке могут встречаться только 3 различных символа, следовательно, при равномерном кодировании, которое требуется по условию задачи, Петя будет тратить на код каждого символа 2 бита и, следовательно, его строка будет занимать в памяти $2*N$ бит. Поскольку в строке может быть только 10 различных троек, на код одной тройки Вася будет тратить 4 бита. Тогда, строка Васи будет занимать в памяти $4*(N/3)$ бит, то есть, $4/3*N$ бит. Легко составить уравнение:

$2*N - 4/3*N = 40$ и получить $N=60$, что и будет правильным ответом.

3. Основы логики (1 балл)

[Эквиваленции]

Известно следующее равенство:

$$((\bar{B} \rightarrow \bar{A}) \rightarrow \bar{B}) \vee (\bar{A} \rightarrow \bar{C}) \wedge \bar{C} \vee \overline{A \rightarrow \bar{D}} = \text{ложь}$$

Тогда для каких из перечисленных логических выражений можно однозначно определить их логическое значение (истинность или ложность)? Отметьте **все** подходящие выражения.

1. $A \wedge B \leftrightarrow C \wedge D$
2. $(A \rightarrow B) \leftrightarrow (C \rightarrow D)$
3. $A \leftrightarrow B \wedge C \wedge D$
4. $A \wedge C \leftrightarrow B \wedge D$
5. $(A \rightarrow C) \leftrightarrow (B \rightarrow D)$
6. $B \leftrightarrow A \wedge C \wedge D$
7. $B \wedge C \leftrightarrow A \wedge D$
8. $C \leftrightarrow A \wedge B \wedge D$
9. $D \leftrightarrow A \wedge B \wedge C$
10. $(D \rightarrow C) \leftrightarrow (A \rightarrow B)$

Ответ: 6, 7, 8, 10

Решение:

Упростим выражение в левой части равенства и получим:

$$\bar{B} \vee \bar{C} \vee A \wedge D = \text{ложь}$$

Следовательно, В может принимать значение только «истина», С – также может принимать значение только «истина», а для А и D существует 3 возможных комбинации: А = «ложь», D = «ложь»; А = «ложь», D = «истина»; А = «истина», D = «ложь».

Для удобства можно составить фрагмент таблицы истинности, включающий те комбинации значений переменных А, В, С и D, при которых равенство будет выполняться:

A	B	C	D
0	1	1	0
0	1	1	1
1	1	1	0

Рассмотрим каждую из предложенных эквиваленций:

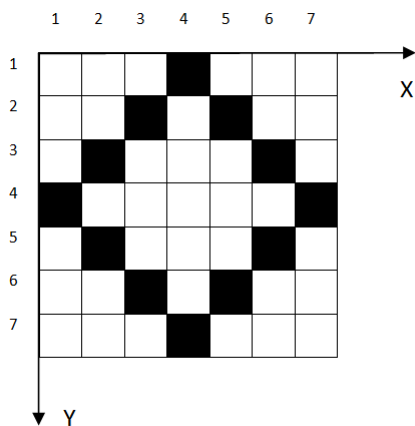
1. Для первой эквиваленции в случае первой комбинации (0110) и левая и правая части будут ложными, следовательно, эквиваленция будет истинна. Для второй строки левая часть эквиваленции будет ложна, а правая истинна, то есть эквиваленция будет ложна. Следовательно, для этого варианта невозможно однозначно установить истинность или ложность.
2. Аналогично вторая эквиваленция будет принимать ложное значение для первой комбинации и истинное для второй.
3. Третья эквиваленция будет принимать истинное значение для первой комбинации и ложное для второй комбинации.
4. Четвертая эквиваленция будет принимать истинное значение для первой комбинации и ложное для второй.
5. Пятая эквиваленция будет принимать ложное значение для первой комбинации и истинное для второй.
6. Для шестой эквиваленции левая часть всегда будет истинна, а правая всегда ложна. Значит мы можем однозначно определить логическое значение шестого выражения – оно будет ложно при всех комбинациях значений переменных, которые мы вывели из исходного логического равенства.
7. Аналогично в седьмой эквиваленции левая часть всегда будет истина, а правая ложна. Следовательно логическое значение этого выражения мы также можем однозначно определить.
8. В восьмой эквиваленции левая часть всегда истина, а правая всегда ложна. Значит и для этого выражения мы можем однозначно определить значение.
9. Девятая эквиваленция будет истинна для первой комбинации и истинна для второй. Не подходит.
10. Последняя, десятая эквиваленция. Левая часть для всех трех комбинаций будет истинна, поскольку в импликации справа истинное значение. Аналогично правая часть также во всех трех случаях будет истинной. Следовательно мы можем однозначно определить значение этого выражения – оно будет истинным.

Таким образом, мы определили, что только для 6, 7 и 8 выражений мы можем однозначно определить их истинность исходя из исходного равенства.

4. Алгоритмизация и программирование. Формальные исполнители (3 балла)

[Рекурсивное закрашивание]

Петя очень давно участвует в олимпиадах и знает, что есть алгоритм рекурсивного закрашивания растрового изображения. У Пети есть множество черно-белых (bitmap) изображений, размером N на N пикселей, где N – нечетное число. На каждом изображении присутствует контур, имеющий форму ромба, центр которого совпадает с центром изображения, а вершины расположены в центрах первой и последней строки, и первого и последнего столбца изображения, соответственно. Линии имеют толщину в один пиксель. Вот пример такого изображения, размером, 7 на 7 пикселей:



Традиционно для компьютерной графики, система координат имеет начало в верхнем левом углу (начальный пиксель имеет координаты (1,1)) и направления осей как показано на рисунке.

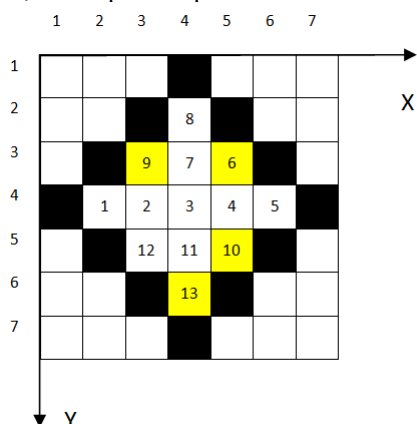
Алгоритм рекурсивного закрашивания заключается в рекурсивном вызове процедуры «Закрасить», которой передаются два параметра – координаты X и Y пикселя.

Процедура *Закрасить*(X,Y), может быть описана следующим образом:

1. Если цвет пикселя с координатами (X,Y) белый, то:
 - a. Изменить цвет пикселя с этими координатами на черный;
 - b. Вызвать процедуру *Закрасить*(X+1,Y);
 - c. Вызвать процедуру *Закрасить*(X,Y-1);
 - d. Вызвать процедуру *Закрасить*(X-1,Y);

2. Иначе завершить процедуру.

Петя решил закрашивать своё изображение, осуществив исходный вызов процедуры Закрасить(2,4) и проанализировал порядок, в котором закрашивались пиксели, выделив пиксели, закрашивание которых произошло в результате прыжков:



Ответ: 12

Для того, чтобы определить количество «прыжков», необходимо аналогично примеру проанализировать порядок закрашивания пикселей. Это можно сделать вручную или написав несложную реализацию определенного в условии рекурсивного алгоритма. Например, так:

```
def plot(x,y):
    if S[x][y]==0:
        global K
        S[x][y]=K
        K+=1
        plot(x,y+1)
        plot(x-1,y)
        plot(x,y-1)
        plot(x+1,y)
```

$$S = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

```
S.append([1])
```

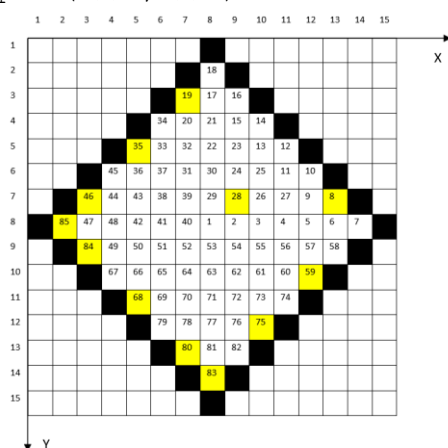
```
for j in range(N):
```

```
S[i].append(0)
```

```
for i in range(N//2+1): #Цикл отрисовки границ закрашиваемой области
```

$$S[N/2+i][i] = -1$$
$$S[N/2-i][i] = -1$$
$$S[i][N//2+i] = -1$$
$$S[N-i-1][N//2+i] = -1$$

```
plot(N//2,N//2)
```

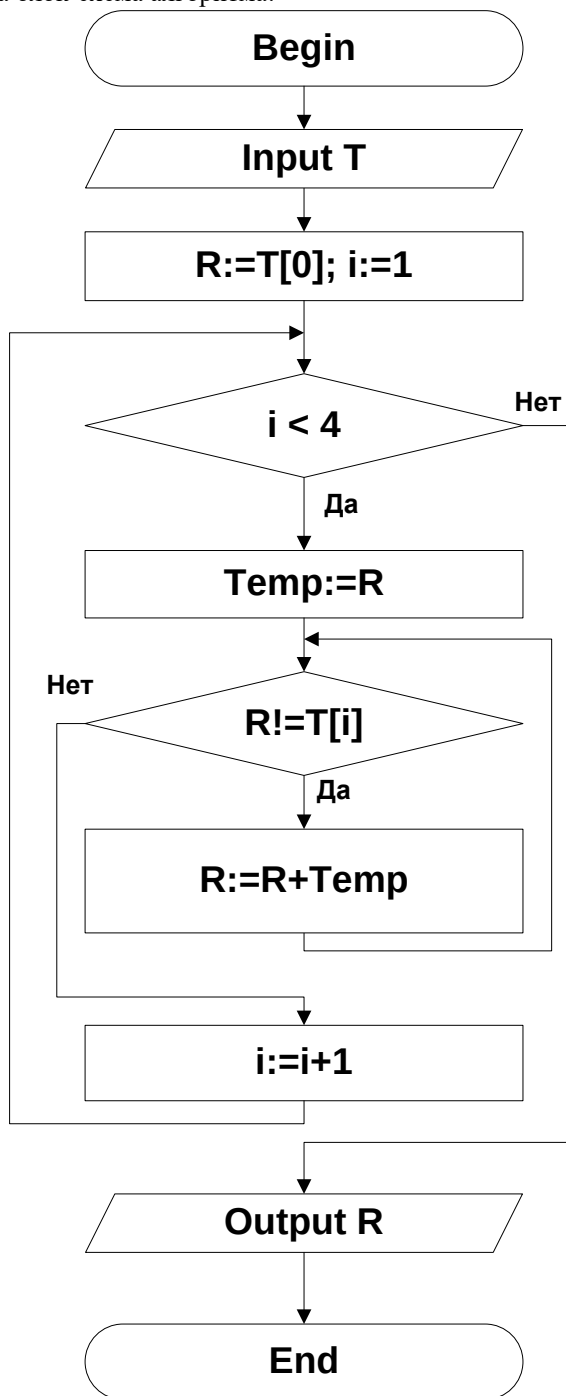


На приведенном рисунке аналогично примеру показан порядок закрашивания пикселей и выделены пиксели, которые были закрашены в результате прыжка. Легко убедиться, что их ровно 12.

5. Алгоритмизация и программирование. Анализ алгоритма, заданного в виде блок-схемы (2 балла)

[Пороги]

Дана блок-схема алгоритма:



На вход подается массив из четырех целых положительных чисел $T=[7, X, Y, Z]$. Все элементы массива попарно различны. Известно, что в результате выполнения алгоритма было выведено число 23023. Определите неизвестные значения элементов входного массива и укажите в ответе через пробел значения первых двух из них: сначала значение X и затем значение Y . Если существует несколько подходящих вариантов, укажите такой вариант, в котором сумма X и Y будет минимальна. Если такого массива не существует, укажите в ответе NULL.

Примечание: нумерация элементов массива идет с нуля, оператор « $\neq$ » означает «нравно».

Ответ: 77 1001

Решение:

Проанализируем алгоритм.

На первом шаге внешнего цикла значение переменной $Temp$ приравнивается к значению переменной R , которая ранее была приравнена к первому элементу массива T , то есть равна 7. Во вложенном цикле это значение будет последовательно суммироваться с переменной R , пока её значение не станет равно первому элементу искомого массива. Таким образом, первый элемент массива равен произведению 7 на некоторое целое число. Обозначим его за A . То есть, первый элемент массива равен

7*А. Затем, на втором шаге внешнего цикла, значение переменной Temp приравнивается к текущему значению переменной R, то есть также становится равно 7*А, и это значение суммируется с переменной R до тех пор, пока результат не станет равен второму элементу массива. Следовательно, второй элемент массива равен произведению 7*А на некоторое целое число. Обозначим его за В. Таким образом, второй элемент массива равен 7*А*В. После этого, на третьем шаге внешнего цикла, значение переменной Temp вновь приравнивается к текущему значению переменной R, то есть также становится равно 7*А*В, и это значение суммируется с переменной R до тех пор, пока результат не станет равен третьему элементу массива. Следовательно, третий элемент массива равен произведению 7*А*В на некоторое целое число. Обозначим его за С. Таким образом, третий элемент массива равен 7*А*В*С. Обратим внимание, что третий элемент массива равен значению, которое будет выведено в результате исполнения алгоритма, поскольку его значение является условием выхода из цикла.

Следовательно, выведенное в результате число 23023=7*А*В*С. Для того, чтобы найти эти значения, разложим число 23023 на простые сомножители. Число 23023=7*11*13*23 является произведением четырех простых чисел. Следовательно, исходя из требования, что все элементы массива Т должны быть различными, а, следовательно, значения А, В и С не могут быть равны 1, значения А, В и С соответствуют значениям 11, 13 и 23 с точностью до перестановки. Обратим внимание, что для того, чтобы сумма первых элементов массива была минимальной, значения А, В и С нужно взять такими, чтобы они были расположены по возрастанию. Следовательно, А=11, В=13, С=23. Тогда Т[0]=7*11=77, Т[1]=7*11*13=1001. Заметим, что тогда Т[2]=7*11*13*23=23023. Таким образом, ответ будет «77 1001».

6. Телекоммуникационные технологии (2 балла).

[EUI-64]

В протоколе IPv6 адрес представляет собой 128 битную двоичную последовательность, записываемую в шестнадцатеричном формате, группами по 4 цифры. Группы разделяются двоеточиями.

В адресе IPv6 есть две логические части:

1) первая часть, называемая префиксом, содержит адрес сети. По умолчанию она начинается с первого (начального) бита адреса и заканчивается на 64-м бите. Однако длина префикса может меняться. Длину префикса в битах указывают вместе с адресом, в виде десятичного числа, записанного через символ “/”

2) вторая часть, начинающаяся сразу после префикса – адрес узла (например, сетевой карты компьютера).

Для удобства IPv6 адрес можно записывать сокращенно для чего:

1) не пишутся ведущие нули в каждой группе

2) не указываются группы, содержащие только «0». Но это делается только для одной последовательности нулевых групп с конца адреса.

Пример:

для адреса

2001:0000:00AF:ABCD:0000:0000:0000:1234 / 64

сокращенная запись будет:

2001:0000:AF:ABCD::1234 / 64

В примере число бит «/64» показывает, что граница префикса проходит по середине адреса, а жирным выделены сокращенные при записи участки.

Часть IPv6 адреса, соответствующая адресу узла редко назначается вручную. Стандартным механизмом формирования адреса служит алгоритм EUI-64 при котором узлу сообщается префикс адреса, а вторую часть адреса узел генерирует сам на основании MAC адреса.

MAC адрес – это аппаратный адрес сетевого устройства, состоящий из 48 бит, записываемых в шестнадцатеричной форме. Адрес имеет свою структуру: первые 24 бита идентифицируют производителя, последние 24 бита назначаются производителем для конкретного устройства. MAC адреса не сокращаются.

MAC адрес, как правило может быть изменен пользователем, но, по стандарту, 7-й с начала адреса бит не может быть выбран произвольно. Действует правило:

1) Если адрес назначен на заводе, то 7-й бит равен 0,

2) если адрес установлен вручную, то 7-й бит равен 1.

Алгоритм EUI-64 сводится к следующим шагам:

1) MAC-адрес делится на две равные части;

2) Между частями вставляется комбинация "FFFF" (так что бы получилось 64-битное значение);

3) Инвертируется 7-й с начала бит получившейся комбинации.

Пусть в консоли маршрутизатора CISCO были набрана команда формирования IPv6 адреса с указанным префиксом по алгоритму EUI-64:

```
INT1(config-if)#ipv6 address 2001:1234:5678:abcd::/64 eui-64
```

После завершения работы команды была выполнена команда получения адресной информации по протоколу IPv6 и получен консольный вывод.

```
INT1#show ipv6 interface fa0/0
FastEthernet0/0 is up, line protocol is up
Global unicast address(es):
2001:1234:5678:ABCD:C000:FF:FE5C:1, subnet is
2001:1234:5678:ABCD::/64 [EUI]
```

Определите исходный MAC адрес конфигурируемого интерфейса и то, как был установлен адрес: установлен вручную или оставлен по умолчанию заводской адрес.

В ответ укажите MAC адрес в шестнадцатеричном виде и, через пробел букву L – в случае, если адрес назначен вручную и G, если адрес оставлен заводской адрес. Например, 001122334455 G.

Ответ: C200005C0001 L

Решение:

Из консольного вывода видно, что интерфейс получил адрес 2001:1234:5678:ABCD:C000:FF:FE5C:1.

Поскольку длина префикса сети указана как /64, собственно адресом узла являются младшие 64 бита, то есть C000:FF:FE5C:1.

Дополним ведущие нули, чтобы получить полную запись адреса: C000:00FF:FE5C:0001.

Поскольку в соответствии с алгоритмом EUI-64 в середину MAC адреса вставляется последовательность FFFE, удалим её из адреса узла и получим C000:005C:0001.

Поскольку MAC адрес записывается без разделителей, уберем двоеточия:

C000005C0001

Третьим шагом алгоритма EUI-64 была инверсия 7-ого бита. Следовательно, нужно провести еще раз инверсию этого бита, чтобы вернуться к его исходному значению. 7-ой бит будет соответствовать 3-ему биту во втором шестнадцатеричном разряде. Таким образом мы заменим первый 0 в адресе на 2 и получим окончательный адрес: C200005C0001.

Теперь остался последний шаг – определить, были ли этот адрес изменен вручную. Поскольку 7-ой бит равен 1, адрес были изменен вручную. Следовательно, в ответ нужно записать «C200005C0001 L».

7. Технологии обработки информации в электронных таблицах, сортировка и фильтрация данных (1 балл)

[5 чисел]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D
1		=СРЗНАЧ(A1:A3)	=МАКС(A1:A3)	
2				
3				
4				
5				
6				

В ячейках диапазона A1:A5 находятся пять неповторяющихся целых положительных чисел. Если выделить диапазон A1:A5 и отсортировать ячейки по убыванию, выяснится, что ячейка B1=18, а ячейка C1=25. Если же отсортировать ячейки этого диапазона по возрастанию, то ячейка B1=10, а ячейка C1=13. Известно, что в одной из ячеек содержится число 9. Определите значения всех ячеек диапазона A1:A5 и запишите в ответ в порядке убывания.

Ответ: 25 16 13 9 8

Решение:

Обозначим неизвестные значения, отсортированные в порядке убывания как x_1, x_2, x_3, x_4 и x_5 соответственно. Рассмотрим результат сортировки ячеек по убыванию. Поскольку ячейка C1 содержит максимальное значение, после сортировки по убыванию, в ней окажется значение x_1 . Так как ячейка B1 в этом случае равна 18, получается, что $\frac{25+x_2+x_3}{3} = 18$, то есть $x_2 + x_3 = 29$.

Теперь рассмотрим сортировку по возрастанию. Поскольку все числа разные, максимальное значение будет соответствовать x_3 , то есть $x_3 = 13$. Тогда из предыдущего равенства $x_2 = 29 - 13 = 16$. Значение B1 после сортировки по возрастанию равно 10, следовательно $\frac{13+x_4+x_5}{3} = 10$, а значит $x_4 + x_5 = 17$. По условию одно из чисел равно 9, значит это x_4 , поскольку x_5 будет $17-9=8$. Теперь остается записать найденные значения в указанном порядке: 25 16 13 9 8

8. Технологии программирования (2 балла)

[Экспресс тест]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

В данной задаче вам предлагается автоматизировать оценку результата экспресс теста.

Вам дана двухцветная картинка размером 10×20 . Для обозначения цветов используются символы «#» и «.». Тест считается отрицательным, если на картинке изображена одна вертикальная полоска и положительным, если три. В любом другом случае тест считается испорченным.

Полоской будем считать область картинки $10 \times k$, состоящую из символов «#», где k может быть произвольным. При этом все соседние клетки с этой областью должны быть «.». Полоска может находиться на границе картинки.

Формат входных данных

В первой строке входных данных задано число t - число тестов ($1 \leq t \leq 30$). В следующих $t \cdot 10 + (t - 1)$ строках заданы картинки тестов. Соседние картинки разделены пустыми строками. После последней картинки, пустой строки нет.

Каждая картинка состоит из 10 строк по 20 символов, каждый из которых либо «#», либо «.».

Формат выходных данных

Для каждой картинки выведите результат теста в отдельной строке:

Negative - если тест отрицательный;

Positive - если тест положительный;

Incorrect - если тест испорчен.

Пример

Стандартный ввод	Стандартный вывод
4##.....##.....##.....##.....##.....##.....##.....##.....##.....##..... ##.....##.....##.....#.....##.....##.....##.....##.....##.....##.....##..... ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..#.....#.....###. ..##.....#.....###. ..##.....#.....###. ..##.....#.....###. ..##.....#.....###. ..#.....#.....###. ..#.....#.....###. ..##.....#.....###. ..#.....#.....###.	Negative Incorrect Positive Incorrect

Решение:

Для решения данной задачи в варианте с вертикальными полосками в каждом тесте достаточно рассмотреть первую строчку. Чтобы тест был корректным должны быть выполнены следующие условия: • Если в первой строке в i -м столбце стоит символ «#», то в остальных строках i -го столбца тоже должны стоять символы «#» • Если в первой строке в i -м столбце стоит символ «.», то в остальных строках i -го столбца тоже должны стоять символы «.» В случае, если вышеуказанные условия выполнены, можно утверждать, что на картинке изображены корректные вертикальные полоски и, возможно, тест не испорчен. Теперь нужно посчитать сколько их. Число полосок на картинке равно числу их левых границ. Левая граница

полоски находится в столбце i , если в i -м столбце первой строки находится символ «#» и либо $i = 1$, либо в $i - 1$ -м столбце первой строки находится символ «.». Таким образом, посчитав число левых границ полосок можно посчитать число полосок. Согласно условию «Тест считается отрицательным, если на картинке изображена одна вертикальная полоска и положительным, если три. В любом другом случае тест считается испорченным». Соответственно, остается только рассмотреть случаи и вывести ответ.

9. Технологии программирования (4 балла)

[Убрать циклы]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

Мальчик Коля любит графы, но не любит циклы. У Коли есть несколько ориентированных не обязательно связанных графов. Графы могут содержать параллельные ребра, но не содержат петли. К сожалению, некоторые из них могут содержать циклы.

Коля хочет развернуть некоторые ребра в каждом графе, чтобы избавиться от циклов. Разворотом ребра будем считать замену ребра из вершины a в вершину b на противоположное ребро, которое будет направлено из вершины b в вершину a . При этом Коля хочет развернуть наименьшее число ребер в каждом графе, чтобы получившиеся графы не содержали циклов.

Помогите Коле узнать, какое наименьшее число ребер ему нужно развернуть в каждом графе.

Формат входных данных

В первой строке входных данных содержится число t - число графов ($1 \leq t \leq 8$).

В первой строке описания каждого графа дано два числа n и m - число вершин и число ребер соответственно ($2 \leq n \leq 10, 1 \leq m \leq 1000$).

В следующих m строках содержатся пары чисел a_i, b_i - описание ребер графа ($1 \leq a_i, b_i \leq n, a_i \neq b_i$).

Для удобства, описания графов разделены пустыми строками. Пустой строки после последнего графа нет.

Гарантируется, что суммарное число вершин во всех графах не превосходит 50.

Формат выходных данных

Для каждого графа выведите одно число в отдельной строке - наименьшее число ребер, которые нужно развернуть, чтобы полученный граф не содержал циклов.

Пример

Стандартный ввод	Стандартный вывод
4	1
2 2	0
1 2	1
2 1	1
4 2	
1 4	
4 3	
3 3	
1 2	
2 3	
3 1	
3 5	
1 2	
2 3	
3 1	
2 3	
1 3	

Замечание

В первом графе нужно развернуть одно из ребер, чтобы убрать цикл. Во втором графе циклов изначально нет, так что ничего разворачивать не надо. В третьем графе можно развернуть любое ребро, чтобы убрать цикл. В четвертом графе нужно развернуть ребро (3,1) или ребро (1,2), чтобы новый граф не содержал циклов.

Решение:

В данной задаче нужно было сделать граф ациклическим путем разворота некоторых ребер. Если граф ациклический, то у него есть топологическая сортировка. Значит, если бы позволяли ограничения, можно было бы перебрать все перестановки вершин графа в качестве топологических сортировок и найти ту, которая получается минимальным числом разворотов ребер. Для того, чтобы узнать для фиксированной топологической сортировки, сколько ребер нужно перевернуть, можно пройти двумя циклами по всем парам вершин и если первая вершина идет в топологической сортировке раньше, чем вторая, то нам необходимо развернуть все ребра из второй вершины в первую, чтобы топологическая сортировка была корректна. Соответственно, сумма таких ребер «справа налево» будет ответом для данной топологической сортировки. Таким образом, минимальным числом ребер, которые нужно развернуть, будет минимум из полученных ответов для всех топологических сортировок.

К сожалению, данное решение работает за $O(n! \cdot n^2)$, что не укладывается ограничения этой задачи, поэтому нужно как-то оптимизировать данное решение. Для оптимизации используем метод динамического программирования по подмножествам.

- $dp[mask]$ — минимальное число ребер, которые нужно развернуть, чтобы граф состоящий из вершин в маске $mask$ стал ациклическим

- В пустом графе ничего разворачивать не нужно, значит $dp[0] = 0$

- $dp[mask] = \min(dp[mask - 2^i] + cnt(i, mask - 2^i))$ для всех i , которые есть в маске. Здесь $cnt(i, mask - 2^i)$ — число ребер, которые направлены из вершины i в вершины, которые есть в маске $mask - 2^i$. Суть данного перехода в том, что мы уже знаем ответ для масок $mask - 2^i$, значит для них существует некоторая топологическая сортировка, к этой топологической сортировке, мы как-будто дописываем в конец вершину i . Данное решение будет работать за $O(2^n \cdot n^2)$, что в свою очередь уложится в необходимые ограничения.

Отборочный этап. Первый тур (приведен один из вариантов заданий)

1. Кодирование информации. Системы счисления (2 балла)

[Четыре-шестнадцать]

Однажды Петя похвастался Васе, что знает, как связаны между собой четверичная и шестнадцатеричная система счисления. В частности, он заявил, что проанализировал все целые положительные числа, меньшие 4096_{10} , и обнаружил среди них число с максимальной разностью между суммой цифр в записи этого числа в шестнадцатеричной системе счисления и суммой цифр в записи этого числа в четверичной системах счисления. Вася немного подумал и сказал, что такое число не одно. Помогите Пете и определите, сколько существует таких чисел. В ответе укажите целое число.

Ответ: 64

2. Кодирование информации. Системы счисления (3 балла)

[Две цифры]

Сколько существует целых положительных чисел таких, что их запись в шестнадцатеричной системе счисления оканчивается на две одинаковые цифры, запись не содержит цифр 0, и сумма цифр в записи не превосходит 10_{10} . В ответе укажите целое число.

Ответ: 341

3. Кодирование информации. Количество информации. Кодирование текста (1 балл)

[Электронный документ]

Петя разрабатывает прототип системы электронного документооборота. Он решил не использовать СУБД, а записывать каждый электронный документ в файл, используя собственный формат. Электронный документ состоит из двух частей: аннотации и основной части. Количество символов в аннотации (N) и в основной части (M) Петя хранит отдельно, поэтому в файл он решил записывать только последовательно коды символов сначала аннотации, а потом основной части. Сжимать данные Петя пока не научился, а из всех кодировок прочитал только про двухбайтную кодировку. Эту кодировку он и решил использовать для записи кодов символов, тем более что тексты документов могут содержать символы различных языков. Вася заметил, что по требованиям к системе документооборота в аннотации к документу возможно использовать только ограниченное количество символов, а именно, строчные и прописные буквы латинского и русского алфавитов, арабские цифры и 30 знаков препинания и специальных символов. Вася рассчитал, что если для аннотации хранить коды символов, используя минимальное, одинаковое для всех символов количество бит, а для основной части по-прежнему использовать двухбайтную кодировку, размер файла уменьшится ровно на $1/8$.

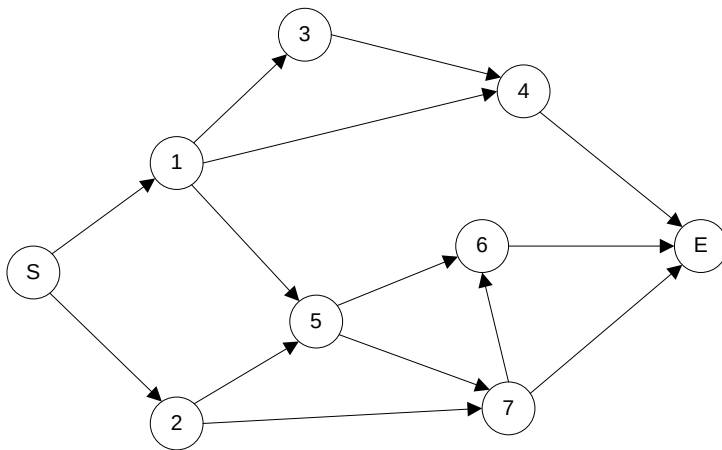
Затем Вася проанализировал требования к основной части документа и заметил, что количество символов в алфавите основной части можно уменьшить в 4 раза по отношению к количеству символов в алфавите, которое может быть при двухбайтной кодировке. Тогда Вася предложил еще уменьшить размер файла за счет того, что для основной части хранить коды символов нового алфавита, используя минимальное, одинаковое для всех символов количество бит. В результате расчетов размер нового файла получился на 3584 байт меньше, чем был у исходного файла Пети. Определите количество символов в аннотации (N) и количество символов в основной части (M). В ответе укажите через пробел два целых числа, сначала N , а затем M .

Ответ: 2048 6144

4. Кодирование информации. Количество информации (1 балл)

[Сложная траектория]

Схема дорог представлена в виде графа:



Петя движется из пункта S в пункт E, проходя в процессе через несколько промежуточных пунктов, обозначенных цифрами от 1 до 7. Стрелки указывают возможные направления движения. В каждом промежуточном пункте, из которого исходит более одной стрелки, Петя с равной вероятностью выбирает любой из доступных способов продолжить путь. Есть несколько сообщений:

1. Петя оказался в пункте 5.
2. Петя оказался в пункте 2.
3. Петя оказался в пункте 7.
4. Петя оказался в пункте 4.
5. Петя оказался в пункте 6.

Необходимо упорядочить эти сообщения по возрастанию количества информации в каждом из них. В ответе укажите подряд 5 чисел – номера сообщений в требуемом порядке.

Ответ: 23514

5. Основы логики. Анализ логических функций (3 балла)

[Логическая цепочка]

Вася сконструировал логический преобразователь. На вход преобразователя подается двоичная последовательность длиной в N бит ($N > 3$). На выходе получается одно логическое значение: истина или ложь. Преобразователь работает по следующему алгоритму:

1. Первый бит последовательности считая слева направо принимается за текущий.
2. Берутся последовательно 3 бита, начиная с текущего. Их значения интерпретируются как значения логических переменных A, B и C соответственно. При этом единичное значение первого бита считается за истинное значение переменной A, а нулевое – за ложное. Аналогично для переменных B и C.
3. Вычисляется значение логической функции: $F(A, B, C) = (A \wedge B \rightarrow C) \wedge (\neg(A \vee \neg B) \rightarrow \neg C)$.
4. Если $F(A, B, C)$ приняло ложное значение, алгоритм завершается со значение «ложь» на выходе.
5. Если $F(A, B, C)$ приняло истинное значение, и значение C соответствовало последнему биту последовательности, алгоритм завершается со значение «истина» на выходе.
6. Если $F(A, B, C)$ приняло истинное значение, и значение C было не последним битом последовательности, номер текущего бита увеличивается на единицу и происходит переход на шаг 2.

Вася перепробовал все последовательности длиной N бит и увидел, что только для 611 из них преобразователь получил значение «истина» на выходе. Определите N и запишите в ответ целое число.

Ответ: 13

6. Основы логики. Упрощение логического выражения (1 балл)

[Цепочка следствий]

Упростите логическое выражение или укажите его результат (при его однозначности). Результат упрощения может содержать только операции инверсии, конъюнкции и дизъюнкции.

$$(((((((A \rightarrow B) \rightarrow A) \rightarrow C) \rightarrow A) \rightarrow D) \rightarrow A) \rightarrow E) \rightarrow A) \rightarrow F$$

Комментарий по вводу ответа: операнды вводятся большими **латинскими** буквами; логические операции обозначаются, соответственно как **not**, **and** и **or**.

Скобки используются только для изменения порядка выполнения операций. Если порядок выполнения операций очевиден из их приоритетов – дополнительное использование скобок считается ошибкой.

При однозначном ответе – истинный ответ обозначается как 1, а ложный как 0.

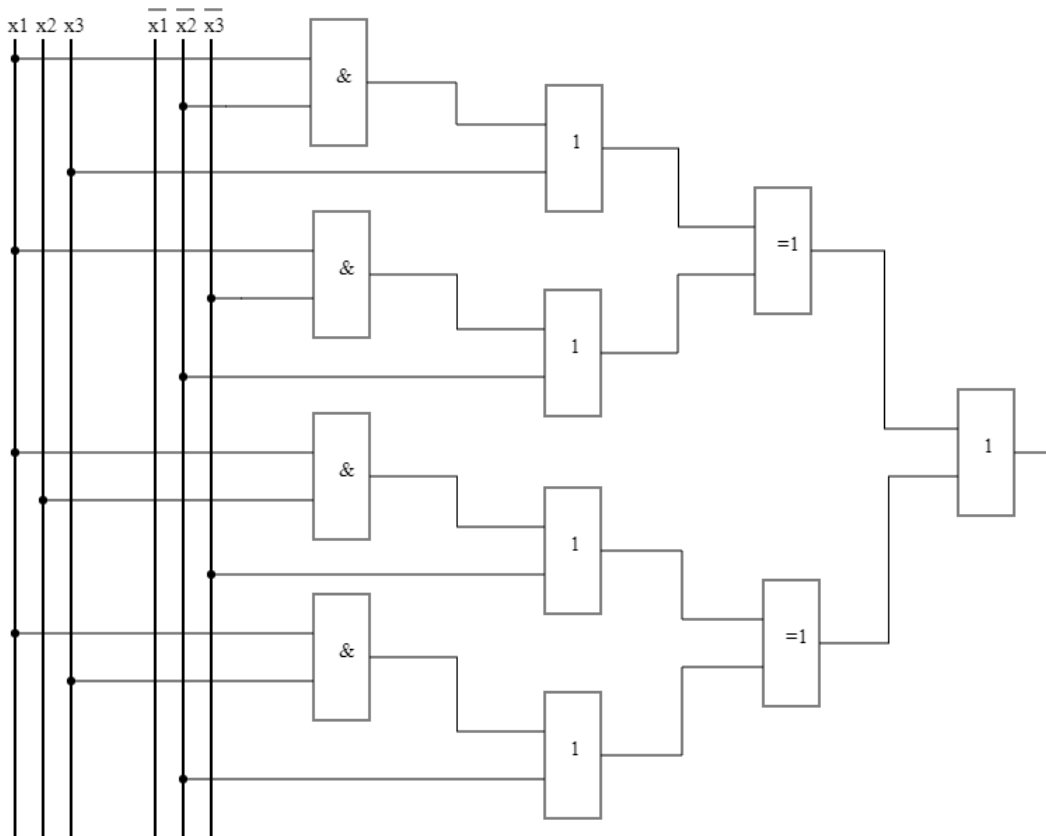
Пример записи ответа: $(A \text{ or not } B) \text{ and } C$

Ответ: not A or F | F or not A

7. Основы логики. Синтез выражения по логической схеме (2 балла)

[Три переменных]

Дана схема логической функции $F(x_1, x_2, x_3)$.



Найдите такую комбинацию значений логических переменных x_1 , x_2 и x_3 , при которой на выходе будет получаться значение «ложь». В ответе укажите подряд три значения 0 или 1, соответствующие значениям логических переменных в порядке возрастания их индексов, где 0 означает ложное значение, а 1 – истинное значение. Если таких комбинаций несколько, укажите любую из них. Если таких комбинаций нет, укажите в ответе NULL. Пример записи ответа: 010.

Примечание. На схеме использованы следующие обозначения логических операторов:

Конъюнкция	Дизъюнкция	Исключающее ИЛИ

Входы с горизонтальной чертой над логической переменной означают её инверсию.

Ответ: 100

8. Алгоритмизация и программирование. Формальный исполнитель (2 балла)

[Калейдоскоп]

Строки обрабатываются в соответствии со следующим алгоритмом:

1. Заменить в строке все символы 'C' на подстроку 'BCB'
2. Заменить в строке все подстроки 'BB' на подстроку 'ABA'
3. Заменить в строке все подстроки 'AA' на подстроку 'ACA'
4. Подсчитать количество символов 'C' в строке. Если оно более 500, завершить исполнение алгоритма, в противном случае перейти на шаг 1.

Пусть изначально была строка 'ABCBA'. Определите символы, которые в строке, получившейся после обработки, находятся на позициях 500, 1001 и 2002. Позиции символов считаются слева направо, начиная с 1. В ответе укажите через пробел три символа: сначала символ на позиции 500, затем символ на позиции 1001 и затем символ на позиции 2002.

Ответ: B A C

9. Алгоритмизация и программирование. Анализ алгоритма (3 балла)

[Спирали]

Петя написал программу, которая по заданному числу N строит квадратную матрицу $N \times N$, заполненную последовательно натуральными числами по часовой стрелке, начиная с верхнего левого угла. Например, для некоторых значений N его программа вывела такие матрицы:

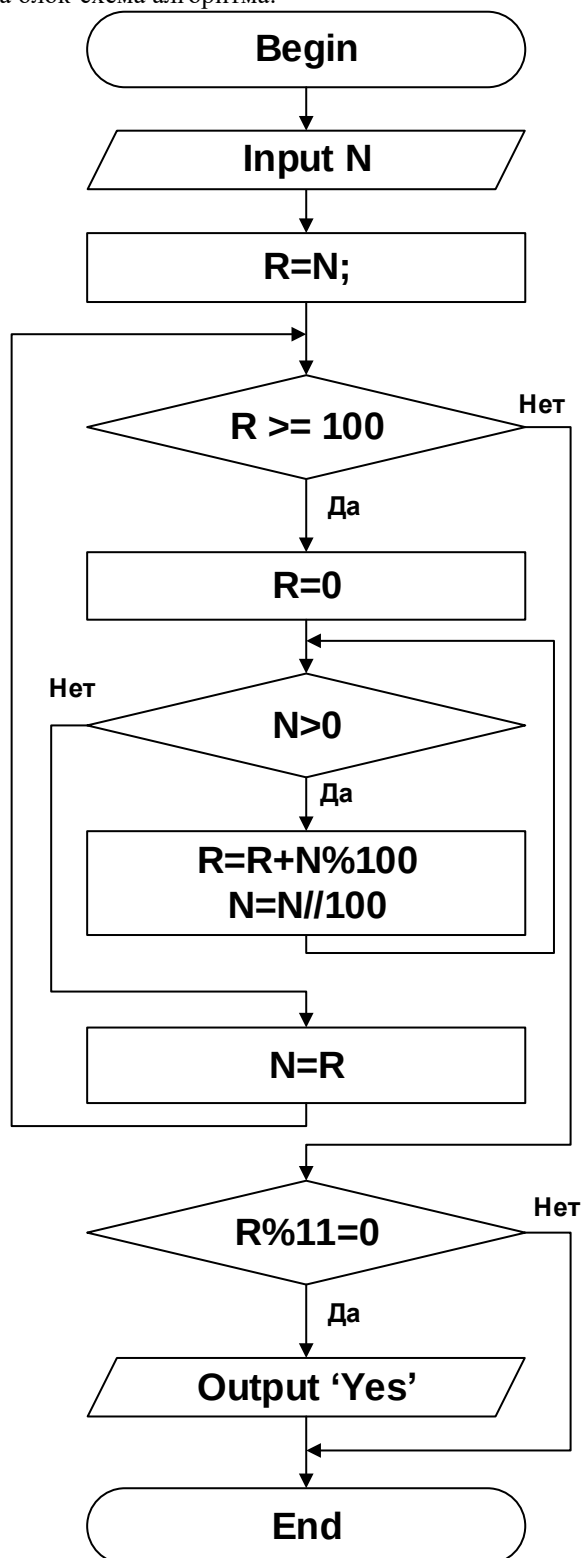
N=4	N=5
1 2 3 4	1 2 3 4 5
12 13 14 5	16 17 18 19 6
11 16 15 6	15 24 25 20 7
10 9 8 7	14 23 22 21 8
	13 12 11 10 9

Вася предложил ввести характеристику R для таких матриц, которая будет равна сумме элементов той строки матрицы, в которой сумма элементов максимальная. Например, для матрицы, построенной при $N=4$, значение R будет равно 48, а при $N=5$ – R примет значение 91. При каком значении N получится значение $R=10581$? В ответе укажите целое число или NULL, если такого значения N не существует.

Ответ: 25

10. Алгоритмизация и программирование. Блок-схема, анализ алгоритма (2 балла) [undecim]

Дана блок-схема алгоритма.



Алгоритм был выполнен для всех значений N из диапазона [1000,1500]. Для скольких значений в результате его выполнения было выведено слово 'Yes'? В ответе укажите целое число.

Примечание. Операция A/B вычисляет частное от целочисленного деления A на B. Операция A%B вычисляет остаток от целочисленного деления A на B.

Ответ: 46

Отборочный этап. Второй тур (приведен один из вариантов заданий)

1. Электронные таблицы. Адресация ячеек и вычисления (1 балл)

[100x100]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D
1	=СУММ(C3:CX102)			
2		0	=B2+\$B\$1	
3		=B2+\$A\$2	=ЕСЛИ(C\$2>\$B3;1;ЕСЛИ(C\$2<\$B3;-1;0))	
4				

Ячейку C2 скопировали во все ячейки диапазона D2:CX2. Ячейку B3 скопировали во все ячейки диапазона B4:B102. Ячейку C3 скопировали во все ячейки диапазона C3:CX102. В ячейку B1 поместили число 73. Какое целое положительное число необходимо поместить в ячейку A2, чтобы в ячейке A1 получилось значение 5127. В ответе укажите целое число.

Ответ: 36

2. Сортировка и фильтрация данных (3 балла)

[Купить или произвести]

База данных системы управления производством состоит из трех таблиц:

Стоимость приобретения деталей	
Название детали	Стоимость приобретения
Деталь 1	137
Деталь 2	214
Деталь 3	4043
Деталь 4	1343
Деталь 5	
Деталь 6	132
Деталь 7	4124
Деталь 8	111
Деталь 9	74

Стоимость сборки деталей	
Название детали	Стоимость сборки
Изделие	142
Деталь 3	63
Деталь 4	29
Деталь 7	38

Состав		
Название собираемой детали	Название требуемой детали	Количество требуемых деталей
Изделие	Деталь 3	2
Изделие	Деталь 7	3
Деталь 3	Деталь 1	1
Деталь 3	Деталь 4	3
Деталь 3	Деталь 6	4
Деталь 4	Деталь 5	4
Деталь 4	Деталь 9	6

Деталь 7	Деталь 2	5
Деталь 7	Деталь 4	2
Деталь 7	Деталь 8	3

Задачей производства является сборка Изделия. Изделие собирается из деталей. Некоторые детали могут быть только приобретены, а некоторые детали могут быть или приобретены или собраны из других деталей. В таблице «Стоимость приобретения деталей» указана стоимость каждой детали в случае её приобретения. В таблице «Состав» указано, сколько каких деталей требуется для сборки одной детали из тех, которые можно собрать, включая само Изделие. В таблице «Стоимость сборки» указана стоимость сборки одной детали, которую можно собрать. При составлении плана производства система для каждой детали, которую можно собрать или приобрести выбирает тот способ получения детали, который будет дешевле. Стоимость детали в случае её сборки складывается из стоимости необходимого количества других деталей и стоимости сборки.

При какой максимальной стоимости «Деталь 5» можно получить Изделие, стоимость которого не превышает 17647?
В ответе укажите целое число.

Ответ: 126

3. Мультимедиа технологии (2 балла)

[Цветовой тон]

Одним из инструментов, позволяющих дать обобщенную характеристику цветов растрового изображения, является гистограмма. Она представляет собой график, по оси абсцисс которого отложены яркости каналов изображения (в нашем случае красного, зеленого и синего соответственно), а по оси ординат количество точек, имеющих такую яркость в выбранном канале.

Одним из способов цветокоррекции изображений является преобразование цветов точек в цветовом пространстве HSB (Hue, Saturation, Brightness).

Пусть дано растровое изображение, содержащее пиксели четырех цветов: (R=250, G=128, B=5), (R=250, G=5, B=5), (R=128,G=250, B=5) и (R=250, G=250, B=5). Изображение содержит равное количество пикселей каждого из этих четырех цветов.

Ниже приведены гистограммы каналов RGB изображения до и после цветокоррекции. Известно, что в процессе цветокоррекции изображение перевели в цветовую модель HSB, изменили значение Hue на +X градусов и перевели обратно в цветовую модель RGB. Выберите из предложенных вариантов значение X, которое было использовано при цветокоррекции.



Варианты ответа:

- 15
- 30
- 45
- 60
- 75
- 90

7. 105
8. 120
9. 135
10. 150
11. 165
12. 180

Ответ: 5

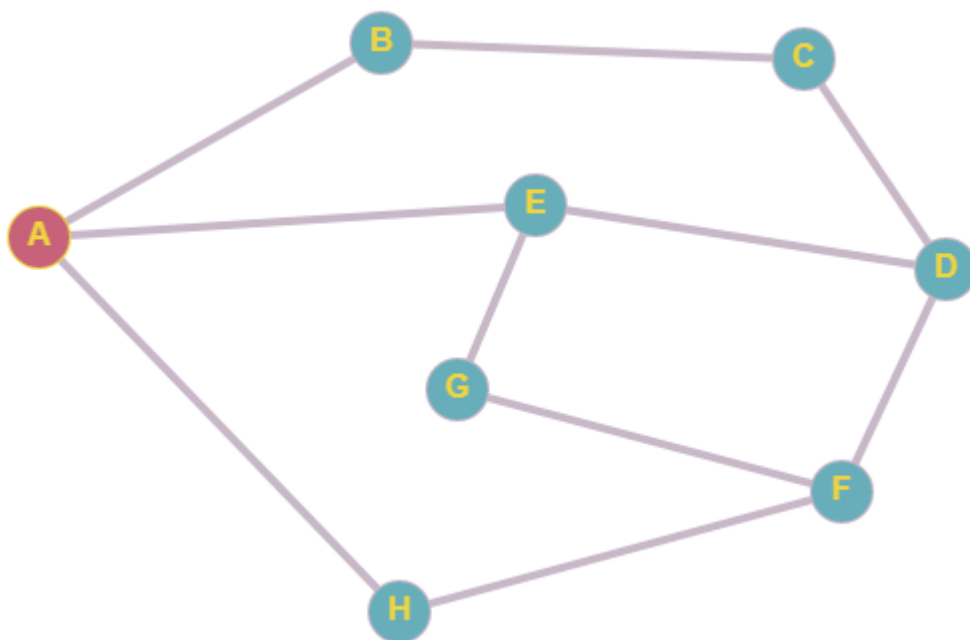
4. Телекоммуникационные технологии (4 балла).

[Управление распределенной памятью]

Распределенные вычислительные системы часто используют для управления распределенной памятью метод «размножения страниц для чтения». В этом случае, если узлу распределенной вычислительной системы требуется страница памяти, которая есть на другом узле, и он планирует использовать её только для чтения данных, ему делается копия этой страницы. Соответственно, со временем, копии одной и той же страницы могут оказаться на значительном количестве узлов. Такой метод позволяет повысить производительность, но несет в себе риск нарушения целостности данных, если один из узлов решит поменять данные на странице. В этом случае для предотвращения появления различающихся по данным копий одной и той же страницы, требуется перед осуществлением модификации данных удалить все её копии в распределенной вычислительной системе. Для этого узел, который решит осуществить модификацию данных на странице, должен послать всем другим узлам уведомление об этом, дождаться подтверждения, что все копии удалены и только после этого начинать модификацию. Сложность в реализации такого решения заключается в том, что исходя из принципов построения распределенных вычислительных систем ни один узел не имеет прямого доступа ко всем остальным узлам и, тем более, не знает о наличии или отсутствии у них копий страниц. Узел может отправлять запросы и получать ответы только от нескольких соседних узлов распределенной вычислительной системы.

Рассмотрим модель.

Пусть есть распределенная вычислительная система из 8 узлов, связанных так, как показано на рисунке:



Известно время в секундах, которое требуется для передачи данных (запроса или отчета) от одного узла к другому:

	A	B	C	D	E	F	G	H
A		2			4			3
B	2		2					
C		2		4				
D			4		4	2		
E	4			4			2	
F				2			2	2
G					2	2		
H	3					2		

Удаление копий реализуется в соответствии со следующими правилами:

1. Узел, который хочет модифицировать страницу, посылает запрос всем узлам, с которыми он связан, и ждет ответа. Запрос имеет поле "Путь". В это поле узел помещает список имен узлов, которым он направил запросы, и свое имя. Это делается для того, чтобы другие узлы не отправляли повторные запросы узлам, про которые уже известно, что им отправлены запросы.

2. Узел, который получил запрос или несколько запросов в определенный момент времени, выполняет следующие действия:
 - а. Получив все запросы, пришедшие в эту секунду, узел проверяет, есть ли связанные с ним узлы, которых нет в полях "Путь" полученных к этому моменту запросов. Если такие узлы есть, он посылает им запрос, в поле "Путь" которого указывает все узлы, которые содержатся в полях "Путь" полученных им запросов и дописывает своё имя и имена узлов, которым рассылает запросы. После этого он ждет от них отчетов.
 - б. Если у узла, получившего запрос, есть страница, которую требуется удалить, узел начинает процедуру её удаления, которая занимает 4 секунды. Рассылка запросов и удаление страницы – независимые операции и начинаются одновременно.
3. Узел, получивший запрос, должен отправить отчет узлу, приславшему запрос. Он сможет это сделать, только когда он дождался отчетов от всех, кому сам послал запрос (если посылал) и когда удалил свою страницу (если она у него была). Если узлу не нужно было удалять страницу, и он не посылал запросы, он отправит ответ мгновенно. Если узел получил несколько запросов, как только у него будет готов отчет на первый из них, он вышлет его всем, от кого получил запросы (повторно запросы другим узлам он не посылает и удаление страницы не прерывает).
4. Если узел получает запрос от узла, которому ранее сам отправлял запрос, то, если у него нет страницы, которую нужно удалять, он мгновенно посылает отчет этому узлу. Если у него есть страница, дожидается её удаления и сразу посылает отчет этому узлу.

Через сколько секунд узел А, который решил модифицировать страницу, получит отчеты на все свои запросы, то есть сможет начать модификацию страницы, если известно, что копии страницы есть на узлах D, E и H? В ответе укажите целое число.

Ответ: 23

5. Операционные системы (2 балла)

[Косвенная адресация]

Одним из распространенных способов организации файловой системы является использование индексных дескрипторов и многоуровневой адресации блоков данных.

Для каждого файла существует индексный дескриптор, который хранится **вне** области данных. Область данных разбита на блоки одинакового размера. Блоки пронумерованы последовательно натуральными числами. Индексный дескриптор хранит некоторые метаданные о файле и 15 номеров блоков из области данных.

Первые 12 номеров являются прямыми адресами – это номера первых 12 блоков с данными файла.

Если размер файла превышает 12 блоков, то 13-ый номер блока в индексном дескрипторе используется для косвенной адресации. Это значит, что он соответствует блоку **в области данных**, но содержимое этого блока будет интерпретировано не как продолжение данных файла, а как последовательность из 32-х битных чисел, являющихся номерами блоков данных, которые уже являются непосредственно данными файла.

Если размер файла настолько большой, что для его хранения не хватит блоков, которые адресуются прямыми адресами и косвенными адресами, используется 14-й номер блока в индексном дескрипторе, реализующий двойную косвенную адресацию. Это значит, что он соответствует блоку **в области данных**, содержимое которого интерпретируется как 32-х битные числа – номера блоков косвенной адресации, также расположенных **в области данных**, которые в свою очередь также состоят из 32-х битных чисел – уже номеров блоков данных файла.

Наконец, для самых больших файлов используется еще один уровень иерархии. 15-ый номер блока в индексном дескрипторе реализует тройную косвенную адресацию. Он соответствует блоку **в области данных**, содержимое которого интерпретируется как 32-х битные числа – номера блоков двойной косвенной адресации, расположенных **в области данных**, содержимое которых интерпретируется как 32-х битные числа – номера блоков косвенной адресации. Они также хранятся **в области данных** и уже их содержимое интерпретируется как 32-х битные числа – номера блоков данных файла.

Естественно, в зависимости от размера файла, количество блоков с адресами различных уровней, хранимых в области данных будет отличаться. При этом, единицей хранения всегда является блок. Поэтому даже, если необходимо хранить всего один адрес, который не поместился в предыдущий блок, будет использован целый блок, содержащий этот адрес.

Таким образом, при использовании многоуровневой адресации блоков данных все номера блоков, кроме 15, хранятся в блоках в области данных.

Пусть используются блоки размером 4 КБайта. Определите, сколько всего блоков будет занято в области данных для хранения файла размером в 17 ГБайт. В ответе укажите целое число.

Примечание:

1 ГБайт = 1024 МБайт; 1 Мбайт = 1024 КБайт; 1 КБайт = 1024 байт.

Ответ: 4460806

6. Технологии программирования (3 балла)

[Подготовка к марафону]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

Мальчик Саша готовится к марафону. Для того, чтобы понять какой результат он должен показать в этом году, Саша изучает результаты прошлого года.

Результаты представлены в виде таблицы, каждая строка таблицы содержит номер контрольной точки, имя участника и время прохождения точки. Первая контрольная точка является стартовой точкой, а последняя контрольная точка является точкой финиша. Все записи отсортированы в хронологическом порядке.

Участники могут стартовать в разное время, но весь марафон проходит в течение одних суток с 00:00:00 по 23:59:59. Не все участники приходят к финишу, некоторые могут сойти в середине дистанции или не успеть на финиш к концу марафона.

Саша хочет выступить не хуже первого участника прошлогоднего марафона. Помогите Саше узнать, какой участник пробежал быстрее всех. Если таких участников несколько, Саша хочет знать всех.

Формат входных данных

В первой строке даны два числа n и m - число контрольных точек и число записей в таблице результатов ($2 \leq n \leq 50; 1 \leq m \leq 1000$).

Далее даны m строк таблицы результатов. В каждой строке через пробел заданы:

Номер контрольной точки p_i ($1 \leq p_i \leq n$). Гарантируется, что все участники проходят точки последовательно без пропусков.

Имя участника, которое задается строкой из латинских букв, первая из которых заглавная, остальные строчные. Длина каждого имени не превосходит 20.

Время прохождения контрольной точки участником в формате hh:mm:ss. Гарантируется, что все записи времени корректны.

Также гарантируется, что все записи отсортированы в хронологическом порядке и каждый участник проходит одну контрольную точку не больше одного раза.

Формат выходных данных

В первой строке выведите число интересующих Сашу участников. Далее выведите имена всех интересующих Сашу участников, по одному в строке в лексикографическом порядке.

Примеры

Стандартный ввод	Стандартный вывод
3 7 1 Vasya 01:11:10 2 Vasya 01:22:23 1 Petya 12:00:00 3 Vasya 12:00:00 2 Petya 13:00:00 3 Petya 14:00:00 1 Kolya 23:59:59	1 Petya
2 4 1 Vasya 00:00:00 2 Vasya 01:00:00 1 Petya 22:00:00 2 Petya 23:00:00	2 Petya Vasya
3 4 1 Vasya 00:00:00 2 Vasya 01:00:00 1 Petya 22:00:00 2 Petya 23:00:00	0
2 6 1 Anya 00:00:00 2 Anya 01:00:00 1 Grisha 15:00:00 2 Grisha 16:00:00 1 Borya 22:00:00 2 Borya 23:00:00	3 Anya Borya Grisha

7. Технологии программирования (5 баллов)

«Простая» задача

Имя входного файла
Имя выходного файла
Ограничение по времени
Ограничение по памяти

стандартный ввод
стандартный вывод
2 секунды
256 мегабайт

В данной задаче по данному массиву a нужно найти число способов выбрать четыре индекса i_1, j_1, i_2, j_2 таких, что $a_{i_1} \cdot a_{j_1} \leq a_{i_2} \cdot a_{j_2}$.

Обратите внимание на ограничения: если просто перебрать числа циклами, вы получите превышение ограничения по времени!

Формат входных данных

В первой строке дано одно натуральное число n - число чисел в массиве ($1 \leq n \leq 10^4$).

Во второй строке заданы n чисел $a_0, a_1, \dots, a_{n-1}$ - элементы массива ($1 \leq a_i \leq 500$).

Формат выходных данных

Выведите одно число - число способов выбрать четыре индекса i_1, j_1, i_2, j_2 таких, что $a_{i_1} \cdot a_{j_1} \leq a_{i_2} \cdot a_{j_2}$.

Примеры

Стандартный ввод	Стандартный вывод
2 2 1	11
1 1	1
3 1 2 1	57
3 2 3 6	48

Замечание

В первом тесте под условие подходит 11 четверок:

№	i_1	j_1	i_2	j_2	a_{i_1}	a_{j_1}	a_{i_2}	a_{j_2}	$a_{i_1} \cdot a_{j_1}$	$a_{i_2} \cdot a_{j_2}$
1	0	0	0	0	2	2	2	2	4	4
2	0	1	0	0	2	1	2	2	2	4
3	0	1	0	1	2	1	2	1	2	2
4	0	1	1	0	2	1	1	2	2	2
5	1	0	0	0	1	2	2	2	2	4
6	1	0	0	1	1	2	2	1	2	2
7	1	0	1	0	1	2	1	2	2	2
8	1	1	0	0	1	1	2	2	1	4
9	1	1	0	1	1	1	2	1	1	2
10	1	1	1	0	1	1	1	2	1	2
11	1	1	1	1	1	1	1	1	1	1

Задания для 9 и 10 класса

Заключительный этап (приведен один из вариантов заданий)

1. Кодирование информации. Системы счисления (2 балла)

[Уравнение с тремя неизвестными]

Дано равенство:

$$2334_X + 14X7_Y = 12Y4_Z$$

X, Y и Z являются целыми положительными числами, которые равны значениям отдельных цифр чисел в равенстве или образуют основания систем счисления, в которых записаны эти числа в равенстве. Найдите такие значения X, Y и Z , что для них равенство будет выполняться, и запишите в ответ через пробел сначала значение X в десятичной системе счисления, затем значение Y в десятичной системе счисления и затем значение Z в десятичной системе счисления. Если таких комбинаций несколько, запишите ту, в которой значение X минимально. Если таких комбинаций не существует, запишите в ответ NULL.

Ответ: 6 9 11

Решение:

Перепишем равенство, используя формулу развернутой записи числа:

$$2 \cdot X^3 + 3 \cdot X^2 + 3 \cdot X + 4 + Y^3 + 4 \cdot Y^2 + X \cdot Y + 7 = Z^3 + 2 \cdot Z^2 + Y \cdot Z + 4$$

Немного упростив, получим:

$$2 \cdot X^3 + 3 \cdot X^2 + 3 \cdot X + Y^3 + 4 \cdot Y^2 + X \cdot Y + 7 = Z^3 + 2 \cdot Z^2 + Y \cdot Z$$

Определим ограничения исходя из используемых в записях чисел цифр: $X > 4, Y > 7, Y > X, Z > Y$.

Аналитическое решение будет, возможно, слишком громоздким, поэтому подходящие значения удобно подобрать программно, например, так:

```
f=0
```

```
i=50809
```

```

while f==0:
    answer=[i//(100**2), (i//100)%100, i%100]
    A1=[2, 3, 3, 4]
    A2=[1, 4, answer[0], 7]
    A3=[1, 2, answer[1], 4]
    R1=A1[0]*answer[0]**3+A1[1]*answer[0]**2+A1[2]*answer[0]+A1[3]
    R2=A2[0]*answer[1]**3+A2[1]*answer[1]**2+A2[2]*answer[1]+A2[3]
    R3=A3[0]*answer[2]**3+A3[1]*answer[2]**2+A3[2]*answer[2]+A3[3]
    if (R1+R2)==R3 and answer[0]>max(A1) and answer[1]>max(A2) and answer[2]>max(A3):
        f=1
        print(answer)
    i+=1

```

Результатом выполнения программы будет [6, 9, 11]. Подставив значения в исходное равенство, легко убедиться в правильности решения.

Другой вариант программного решения:

```

from itertools import product

digits = [x for x in range(5, 37)]
for x, y, z in product(digits, repeat=3):
    if 7 < y < z and x < y and \
        2 * x ** 3 + 3 * x ** 2 + 3 * x + 4 + y ** 3 + 4 * y ** 2 + x * y + 7 == z ** 3 + 2 * z ** 2 + y * z + 4:
        print(x, y, z)
        break

```

2. Кодирование информации. Объем информации (2 балла)

[100 картинок]

У Пети есть 100 квадратных растровых изображений со стороной в N пикселей каждое. Петя хранит каждое изображение в виде последовательности кодов оттенков пикселей, используя стандартную цветовую модель TrueColor, то есть, затрачивая для хранения отдельного кода 24 бита. Вася решил помочь Пете уменьшить хранимый объем данных. Он обратил внимание, что все изображения можно разбить на три группы. В первую группу попала ровно половина исходных изображений. В этой группе в каждом изображении встречается только 65536 различных оттенков. Во второй группе ровно четверть исходных изображений. В этой группе в каждом изображении встречается только 16384 различных оттенков. В третью группу попала оставшаяся четверть исходных изображений и в ней в каждом изображении встречается только 1024 различных оттенка. Вася решил хранить изображения следующим образом. Сначала в каждом изображении он хранит его палитру – последовательность из 24-х битных кодов TrueColor такой длины, сколько различных оттенков встречается в соответствующем изображении. Затем он хранит коды для каждого пикселя, определяющие номер оттенка в хранимой палитре так, что для каждого кода используется минимальное, одинаковое для всех кодов в палитре данного изображения количество бит. Вася выяснил, что всего он сэкономил 117125 КБайт данных на всем наборе изображений Пети. Определите N , при котором это возможно.

В ответе укажите целое число.

Примечание. 1 КБайт = 1024 байт.

Ответ: 1024

Решение:

Составим уравнение по условию:

$$100 \cdot N^2 \cdot 24 - 50 \cdot (2^{16} \cdot 24 + N^2 \cdot \log_2 65536) - 25 \cdot (2^{14} \cdot 24 + N^2 \cdot \log_2 16384) - 25 \cdot (2^{10} \cdot 24 + N^2 \cdot \log_2 1024) = 117125 \cdot 2^{10} \cdot 8$$

Осуществим алгебраические преобразования и вычисления:

$$2400 \cdot N^2 - 800 \cdot N^2 - 350 \cdot N^2 - 250 \cdot N^2 = 117125 \cdot 2^{13} + 9600 \cdot 2^{13} + 1200 \cdot 2^{13} + 75 \cdot 2^{13}$$

$$1000 \cdot N^2 = 128000 \cdot 2^{13}$$

$$N^2 = 2^{20}$$

$$N = 2^{10} = 1024$$

Это и будет правильным ответом.

3. Основы логики (3 балла)

[Исключающие ИЛИ]

Дано логическое равенство:

$$(A \text{ xor } B) \rightarrow (A \text{ xor } F(A, B)) = F(A, B) \rightarrow (B \text{ xor } A)$$

Найдите логическую функцию $F(A, B)$, такую, что указанное равенство будет выполняться.

Если таких функций несколько – запишите любую из них.

В ответе запишите формулу, которая может содержать логические переменные A и B и не более чем три логические операции. Если таких функций не существует, запишите в ответ NULL.

Комментарий по вводу ответа: операнды вводятся большими латинскими буквами; логические операции обозначаются, соответственно как not, and и or. Запись не должна содержать скобок. Пример записи ответа: A or not B

Ответ: not A and B || B and not A

Решение:

Поскольку $F(A, B)$ является функцией от двух логических аргументов, построим её таблицу истинности. Для этого последовательно подставим в логическое равенство четыре возможные комбинации значений A и B .

При $A=0$ и $B=0$ получим:

$$(0 \text{ xor } 0) \rightarrow (0 \text{ xor } F(A, B)) = F(A, B) \rightarrow (0 \text{ xor } 0)$$

Вычислим скобки, в которых только константы:

$$0 \rightarrow (0 \text{ xor } F(A,B)) = F(A,B) \rightarrow 0$$

Упростим левую и правую часть, раскрыв импликации:

$$1 = \text{not } F(A,B)$$

Следовательно, при $A=0$ и $B=0$, $F(A,B)=0$

При $A=0$ и $B=1$ получим:

$$(0 \text{ xor } 1) \rightarrow (0 \text{ xor } F(A,B)) = F(A,B) \rightarrow (1 \text{ xor } 0)$$

Вычислим скобки, в которых только константы:

$$1 \rightarrow (0 \text{ xor } F(A,B)) = F(A,B) \rightarrow 1$$

Упростим левую и правую часть, раскрыв импликации:

$$0 \text{ xor } F(A,B) = 1$$

$$F(A,B) = 1$$

Следовательно, при $A=0$ и $B=1$, $F(A,B)=1$

При $A=1$ и $B=0$ получим:

$$(1 \text{ xor } 0) \rightarrow (1 \text{ xor } F(A,B)) = F(A,B) \rightarrow (0 \text{ xor } 1)$$

Вычислим скобки, в которых только константы:

$$1 \rightarrow (1 \text{ xor } F(A,B)) = F(A,B) \rightarrow 1$$

Упростим левую и правую часть, раскрыв импликации:

$$1 \text{ xor } F(A,B) = 1$$

$$\text{not } F(A,B) = 1$$

Следовательно, при $A=1$ и $B=0$, $F(A,B)=0$

При $A=1$ и $B=1$ получим:

$$(1 \text{ xor } 1) \rightarrow (1 \text{ xor } F(A,B)) = F(A,B) \rightarrow (1 \text{ xor } 1)$$

Вычислим скобки, в которых только константы:

$$0 \rightarrow (1 \text{ xor } F(A,B)) = F(A,B) \rightarrow 0$$

Упростим левую и правую часть, раскрыв импликации:

$$1 = \text{not } F(A,B)$$

Следовательно, при $A=1$ и $B=1$, $F(A,B)=0$

Таким образом, мы получили значения $F(A,B)$ для всех возможных комбинаций значений аргументов и можем построить таблицу истинности:

A	B	F(A,B)
0	0	0
0	1	1
1	0	0
1	1	0

Эта таблица истинности содержит только одну строку с истинным значением, следовательно искомая логическая функция является конъюнкцией. Поскольку истинное значение она принимает при $A=0$ и $B=1$, это функция **not A and B**.

Другой вариант решения

Обозначим операцию **xor** как $\oplus$, а искомую функцию $F(A,B) = X$. Тогда наше уравнение будет иметь вид:

$$(A \oplus B) \rightarrow (A \oplus X) = X \rightarrow (A \oplus B)$$

По сути, мы имеем левую и правую части нашего равенства, которые представляют собой две логические функции от трех переменных (A, B, X) . Построим таблицы истинности этих функций, чтобы найти наборы переменных, на которых эти функции равны между собой.

A	B	X	$(A \oplus B)$	$(A \oplus X)$	$(A \oplus B) \rightarrow (A \oplus X)$	$X \rightarrow (A \oplus B)$
0	0	0	0	0	1	1
0	0	1	0	1	1	0
0	1	0	1	0	0	1
0	1	1	1	1	1	1
1	0	0	1	1	1	1
1	0	1	1	0	0	1
1	1	0	0	1	1	1
1	1	1	0	0	1	0

Выберем те строки в этой таблице, в которых значения левой и правой части равенства совпадают.

A	B	X	$(A \oplus B)$	$(A \oplus X)$	$(A \oplus B) \rightarrow (A \oplus X)$	$X \rightarrow (A \oplus B)$
0	0	0	0	0	1	1
0	0	1	0	1	1	0
0	1	0	1	0	0	1
0	1	1	1	1	1	1
1	0	0	1	1	1	1

1	0	1	1	0	0	1
1	1	0	0	1	1	1
1	1	1	0	0	1	0

Выпишем эти строки, вспомнив, что $F(A, B) = X$

A	B	$F(A, B) = X$
0	0	0
0	1	1
1	0	0
1	1	0

Очевидно, что $F(A, B) = \neg A \wedge B$

4. Алгоритмизация и программирование. Формальные исполнители (2 балла)

[Сходящаяся последовательность]

Дан алгоритм обработки числовой последовательности, на вход которому подали возрастающую последовательность натуральных чисел от 1 до N:

1. Если первый элемент последовательности – нечетное число, то из последнего элемента последовательности вычесть значение первого элемента последовательности, в противном случае, к последнему элементу последовательности прибавить значение первого элемента последовательности.
2. Удалить первый элемент последовательности.
3. Если в последовательности осталось более одного элемента, перейти на шаг 1, иначе завершить работу алгоритма и вывести получившуюся последовательность.

Например, для N=5 получится следующая цепочка преобразований:

$[1, 2, 3, 4, 5] \rightarrow [2, 3, 4, 4] \rightarrow [3, 4, 6] \rightarrow [4, 3] \rightarrow [7]$ (последнее значение будет выведено как результат работы алгоритма).

Легко заметить, что результатом работы алгоритма всегда будет последовательность, состоящая из одного элемента.

Найдите все такие N, при которых в результате работы алгоритма этот единственный элемент последовательности будет иметь значение 67. В ответе укажите все подходящие значения N через пробел в порядке возрастания. Если таких значений больше 5, укажите первые пять из них. Если таких значений не существует, укажите в ответ NULL.

Ответ: 45 134

Решение:

Рассмотрим примеры построения последовательностей для нескольких первых значений N, начиная с 3:

$[1, 2, 3] \rightarrow [2, 2] \rightarrow [4]$

$[1, 2, 3, 4] \rightarrow [2, 3, 3] \rightarrow [3, 5] \rightarrow [2]$

$[1, 2, 3, 4, 5] \rightarrow [2, 3, 4, 4] \rightarrow [3, 4, 6] \rightarrow [4, 3] \rightarrow [7]$

$[1, 2, 3, 4, 5, 6] \rightarrow [2, 3, 4, 5, 5] \rightarrow [3, 4, 5, 7] \rightarrow [4, 5, 4] \rightarrow [5, 8] \rightarrow [3]$

$[1, 2, 3, 4, 5, 6, 7] \rightarrow [2, 3, 4, 5, 6, 6] \rightarrow [3, 4, 5, 6, 8] \rightarrow [4, 5, 6, 5] \rightarrow [5, 6, 9] \rightarrow [6, 4] \rightarrow [10]$

$[1, 2, 3, 4, 5, 6, 7, 8] \rightarrow [2, 3, 4, 5, 6, 7, 7] \rightarrow [3, 4, 5, 6, 7, 9] \rightarrow [4, 5, 6, 7, 6] \rightarrow [5, 6, 7, 10] \rightarrow [6, 7, 5] \rightarrow [7, 11] \rightarrow [4]$

Заметим закономерность. Если N – четное число, результирующий элемент последовательности будет равен N/2. Если же N – нечетное число, то к значению, полученному для предыдущего нечетного N, прибавляется 3, то есть это значение может быть выражено как $(3 \cdot N - 1) / 2$.

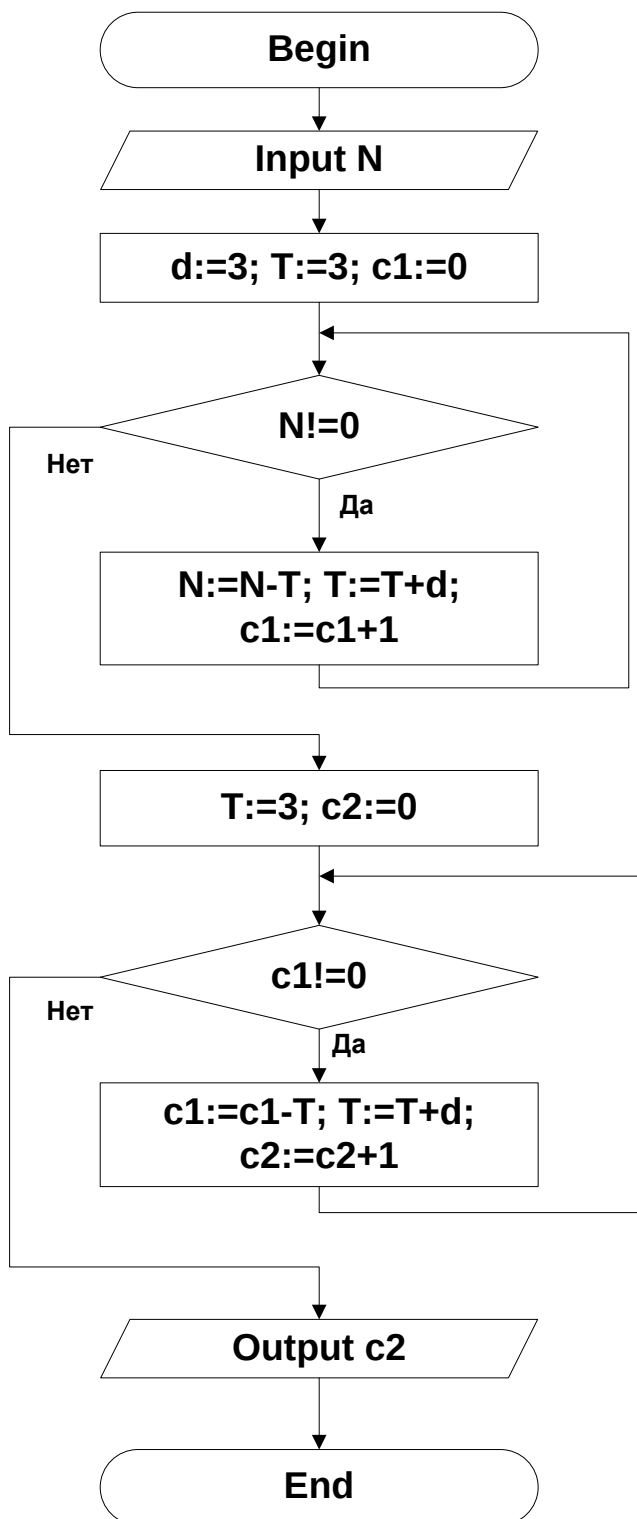
Таким образом, может быть максимум два значения N – четное и нечетное.

Для четного значения N может быть получено умножением данного в условии значения последнего элемента на два, то есть $67 \cdot 2 = 134$. Для нечетного значения N можно рассчитать по формуле $(67 \cdot 2 + 1) / 3 = 45$. Остается записать полученные значения в порядке возрастания и получить ответ «45 134»

5. Алгоритмизация и программирование. Анализ алгоритма, заданного в виде блок-схемы (1 балл)

[До нуля]

Дана блок-схема алгоритма:



Какое целое положительное число нужно подать на вход, чтобы на выходе получилось значение число 18? В ответе укажите целое число.

Примечание. Оператор «!=» означает «не равно».

Ответ: 395523

Решение:

Рассмотрим второй цикл. В этом цикле переменная $c1$ уменьшается на каждом шаге на значение, равное переменной T , которая, в свою очередь, исходно равна 3 и затем увеличивается на 3 на каждом шаге. Выход из цикла возможен, только если переменная $c1$ станет строго равна нулю. Обратим внимание на переменную $c2$. Она представляет собой счетчик шагов этого цикла. Следовательно, успешный выход из цикла за $c2$ шагов означает, что значение $c1$ перед началом цикла было суммой арифметической прогрессии с первым элементом, равным 3, с шагом 3 и длиной в 18 элементов. Воспользуемся формулой для вычисления суммы первых n членов арифметической прогрессии $c1 = (2 \cdot 3 + 3 \cdot (18 - 1)) / 2 \cdot 18 = 513$.

Заметим, что первый цикл устроен точно также. Только уменьшается до нуля введенная на входе переменная N , а начальное значение T и шаг остаются такими же. При этом $c1$ является счетчиком количества шагов в первом цикле, то есть определяет количество элементов в другой арифметической прогрессии, сумме членов которой и должно равняться N . Тогда можно вычислить N по формуле: $N = (2 \cdot 3 + 3 \cdot (513 - 1)) / 2 \cdot 513 = 395523$. Это и будет ответом на задание.

6. Телекоммуникационные технологии (2 балла).

[Приоритизация трафика]

Два приложения – А и В – передают данные по сети используя один неразделяемый канал передачи данных со скоростью 4096 бит в секунду. Данные передаются пакетами. Приложение А передает данные пакетами, размером в 2 КБайт каждый, а приложение В передает данные пакетами, размером, в 6 КБайт каждый. У каждого приложения есть буфер. Приложение А создает и помещает в свой буфер пакет в начальный момент времени и далее по истечении каждой 5-ой секунды. Приложение В создает и помещает в свой буфер пакет по истечении 11-ой секунды от начального момента времени и далее по истечении каждой 20-ой секунды от момента создания первого пакета. Например, если считать начальным моментом времени секунду с номером 0, то приложение А будет помещать пакеты в свой буфер в начале секунд с номерами: 0, 5, 10, 15, 20, ..., а приложение В в начале секунд с номерами: 11, 31, 51, 71, ...

Передача любого пакета происходит целиком, и не может быть прервана. Как только передача пакета завершена, производится проверка наличия готовых к передаче пакетов в буферах приложений. Если хотя бы в одном буфере есть готовый к передаче пакет, в том числе появившийся в буфере в этот момент времени, незамедлительно начинается передача очередного пакета. У приложения В есть приоритет передачи данных по отношению к приложению А. Это означает, что, если есть готовые к передаче пакеты в его буфере, его пакет будет выбран для передачи. Пакет из буфера приложения А будет выбран для передачи только, если в момент выбора нет готовых к передаче пакетов в буфере приложения В. Пакет удаляется из буфера, как только закончена его передача. Какое количество пакетов будет в буфере приложения А в момент окончания передачи 31-го пакета приложения В? В ответе укажите целое число. Если в указанный момент времени в буфере приложения А не будет пакетов, укажите в ответ 0.

Ответ: 63

Решение:

Построим диаграмму, раскрывающую первые 25 секунд описанного процесса:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Появление пакета А	1					2					3					4					5					6
Появление пакета В												1														
Передача пакета А	1					2					3															
Передача пакета В															1											
Пакетов в буфере А	1				0	1				0	1				0	1					2					3
Пакетов в буфере В												1														

В первой строке указаны секунды. Во второй и третьей строках – номера появляющихся пакетов у приложений А и В соответственно. В 5 и 6 строках отмечены периоды передачи пакетов и номера передаваемых пакетов. В 8 и 9 строке указано количество пакетов в буферах приложений в те секунды, когда эти значения изменяются.

Заметим, что начиная с 14 секунды передача данных всегда будет вестись непрерывно, поскольку у приложения А непустой буфер и он не будет успевать его опустошать до появления нового пакета в буфере приложения В с учетом появления новых пакетов в буфере приложения А. Проанализируем следующие 40 секунд:

26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65					
				7					8					9					10					11				12					13					14						
				2																					3																			
4				5				2												6				7				3																
								2																								3												
				3					2	3					4					5					5					4	5					6					7			
0				1															0						1																			

Заметим, что приложение А успевает передать два пакета, после чего начинается передача более приоритетного пакета приложения В. Причем, интервалы образуют период: очередной пакет приложения В появляется на второй секунде передачи второго из двух передаваемых пакетов приложения А. По окончании передачи второго пакета приложения А начинается передача пакета приложения В. За время его передачи в буфере приложения А появляется три новых пакета, плюс еще один пакет в буфере приложения А появляется пока оно передает свои два пакета паузу до появления очередного пакета приложения В. Затем последовательность из передачи двух пакетов приложения А и одного пакета приложения В повторяется с такими же моментами появления очередных пакетов.

Легко сделать вывод, что к моменту окончания передачи очередного пакета приложения В, в буфере приложения А всегда оказывается на два пакета больше, чем в момент окончания передачи предыдущего пакета приложения В. Исходя из того, что к моменту окончания передачи первого пакета приложения В, в буфере приложения А находилось 3 пакета, полученную закономерность можно описать формулой $R = N * 2 + 1$, где N – номер пакета приложения В, а R – искомое количество пакетов приложения А, находящихся в его буфере в момент окончания передачи этого пакета приложения В. Таким образом, в момент окончания передачи 31-го пакета приложения В в буфере приложения А будет $31 * 2 + 1 = 63$ пакета, что и будет правильным ответом.

Программное решение

ah = 5 # Диапазон появления пакетов А
bh = 20 # Диапазон появления пакетов В


```

ta = 2 * 2 ** 13 // 4096 # Время передачи пакета A
tb = 6 * 2 ** 13 // 4096 # Время передачи пакета B
a = 0 # Количество пакетов в буфере A
b = 0 # Количество пакетов в буфере B
ib = 0 # Счетчик пакетов B
t = 0 # Счетчик времени
work = 0 # Обратный отсчет времени передачи очередного пакета
flag = None # Сигнал, какой пакет передается в данный момент
while True:
    if t % ah == 0: # Появляется новый пакет A
        a += 1
    if (t - 11) % bh == 0: # Появляется новый пакет B
        b += 1
    if b != 0 and work == 0: # Если буфер B не пуст и закончена передача очередного пакета
        work = tb # начинаем передавать пакет B
        flag = 'B'
        ib += 1
    elif a != 0 and work == 0: # Если буфер B пуст, а буфер A не пуст и закончена передача очередного пакета
        work = ta # начинаем передавать пакет A
        flag = 'A'
    if work > 0:
        work -= 1
        if work == 0: # Если окончена передача пакета, выясняем какой пакет передавался и удаляем его из буфера
            if flag == 'A':
                a -= 1
            else:
                b -= 1
            if ib == 31: # Если закончен передаваться 31-й пакет B, то выводим содержимое буфера A и
останавливаемся
                print(a)
                break
    t += 1

```

7. Технологии обработки информации в электронных таблицах, технологии сортировки и фильтрации данных (2 балла)

[Три столбца]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D
1				
2	0	=ОСТАТ(ЧАСТНОЕ(A\$1;СТЕПЕНЬ(B\$1;A2));B\$1)	=B2	
3	1		=ЕСЛИ(B2<B3;C2+B3;B3)	
4	2			
5	3			
6	4			
7	5			
8				
9			=СУММ(C2:C7)	
10				

Ячейку B2 скопировали во все ячейки диапазона B3:B7. Ячейку C3 скопировали во все ячейки диапазона C4:C7. В ячейку A1 поместили некоторое целое положительное число, а в ячейку B1 поместили число 8.

Какое максимальное число может быть получено в ячейке C9? Какое минимальное число нужно поместить в ячейку A1 для того, чтобы получить такое значение в ячейке C9? В ответе укажите через пробел два числа: сначала ответ на первый вопрос и затем ответ на второй вопрос.

Ответ: 77 256794

Решение:

Обратим внимание, что формула, записанная в ячейке B2 и скопированная в ячейки диапазона B3:B7 позволяет получить в ячейках диапазона B2:B7 младшие 6 разрядов восьмеричной записи числа, помещенного в ячейку A1. Причем разряды сверху вниз упорядочены от младшего к старшему. Формула, помещенная в ячейку C2 и скопированная в ячейки диапазона C3:C7 считает суммы ячеек диапазона B2:B7, которые расположены по возрастанию.

Соответственно, нам нужно получить максимальную сумму таких сумм, построенных по разрядам числа в восьмеричной системе счисления.

Легко заметить, что максимальную сумму мы получим, если будем максимально долго накапливать сумму в ячейках диапазона C2:C7, то есть если в ячейках диапазона B2:B7 буду подряд расположенные по возрастанию цифры вплоть до максимально возможной в восьмеричной системе счисления цифры 7. Тогда сумма ячеек диапазона C2:C7 составит $2+(2+3)+(2+3+4)+(2+3+4+5)+(2+3+4+5+6)+(2+3+4+5+6+7)=77$. Следовательно, минимальное число, которое нужно поместить в ячейку A1, чтобы получить такую сумму, это десятичное число, соответствующее восьмеричному числу

765432₈. Переведем его в десятичную систему счисления и получим 256794. Запишем ответы в требуемом порядке и получим «77 256794»

8. Технологии программирования (2 балла)

[Финальная стоимость]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

Представьте, что вас наняли в IT отдел сетевого продуктового магазина. Со следующего месяца планируется ввести скидку в 20% на некоторые товары. От вас требуется написать программу расчета финальной стоимости покупки с учетом всех скидок.

Покупка задается в виде списка товаров, где каждый товар описывается в формате <НазваниеТовара> <цена> x<количество>. Если название товара заканчивается на подстроку «sale», то цена на товар при финальном расчете покупки уменьшается на 20% с округлением до целого числа в меньшую сторону. Так, например, если цена товара была 12 рублей, с учетом скидки она составит 9 рублей. На остальные товары скидка не распространяется.

Суммарная стоимость каждого товара вычисляется по формуле $\text{finalPrice} \cdot x$, где finalPrice - финальная цена, возможно, с учетом скидки и округлением, а x - количество товара в списке покупок.

Ваша задача - вычислить итоговую стоимость покупки.

Формат входных данных

В первой строке дано число n - число наименований товаров в покупке ($1 \leq n \leq 1000$).

В следующих n строках заданы описания товаров в покупке в формате: <НазваниеТовара> <цена> x<количество>. Название товара состоит из не более чем 20 строчных латинских букв. Цена - натуральное число, которое строго больше 1, но не больше 1000. Количество - натуральное число не больше 100.

Формат выходных данных

Выведите одно число - итоговую стоимость покупки.

Пример

Стандартный ввод	Стандартный вывод
5 bananassale 10 x2 breadfixed 5 x5 colafixed 20 x2 gumsale 9 x10 tea 2 x15	181

Замечание

В тесте из примера итоговые стоимости товаров составят: 8, 5, 20, 7 и 2 соответственно. Итоговая стоимость покупки: $8 \cdot 2 + 5 \cdot 5 + 20 \cdot 2 + 7 \cdot 10 + 2 \cdot 15 = 181$.

Решение:

В данной задаче нужно было аккуратно написать ровно то, что просилось в условии. Распарсить входные данные после чего либо проверить, что последние четыре буквы названия товара составляют подстроку «sale», либо что последние пять букв названия товара составляют подстроку «fixed». Далее необходимо прибавить к ответу либо $\text{price} \cdot x$, либо $(\text{price} \cdot 4 \div 5) \cdot x$ в зависимости от условия задачи. Здесь price — цена товара в списке покупок, x — количество товара, а $\div$ — операция целочисленного деления. Также корректно домножать price на 0.8, после чего приводить результат к целочисленному типу данных.

9. Технологии программирования (4 балла)

[Напитки и стаканы]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

К Коле на день рождения придут n друзей. По этому случаю он заготовил n бутылок различных напитков, а каждый из друзей принесет свой стакан.

Коля знает размеры стаканов каждого друга, однако, может случиться так, что содержимое конкретной бутылки может не поместиться полностью в конкретный стакан, если объем бутылки больше объема стакана. При этом Коля не хочет, чтобы после угощения что-то осталось, поэтому будет разливать напитки только так, чтобы они полностью поместились в стаканы. Напиток полностью помещается в стакан, если объем, содержащей его бутылки, не превосходит объем стакана.

Коля хочет своеобразно оценить сколько способов есть разлить напитки по стаканам друзей так, чтобы все напитки полностью поместились в стаканы. Так как число способов может быть довольно большим, Коля хочет знать его по модулю 1 000 000 007. Помогите Коле посчитать это число.

Обратите внимание на ограничения, если написать неэффективное решение, например перебор, вы получите превышение ограничения по времени!

Формат входных данных

В первой строке дано число t - число тестовых наборов ($1 \leq t \leq 100$).

Каждый тестовый набор задается тремя строками. В первой из них дано число p_i - число бутылок напитков и стаканов в i -м наборе ($1 \leq p_i \leq 10^5$).

Во второй строке заданы p_i чисел $a_{i,j}$ - объемы бутылок в i -м наборе.

В третьей строке даны p_i чисел $b_{i,j}$ - объемы стаканов в i -м наборе ($1 \leq a_{i,j}, b_{i,j} \leq 100$).

Гарантируется, что сумма p_i по всем тестовым наборам не превосходит $3 \cdot 10^5$.

Формат выходных данных

Для каждого тестового набора выведите одно число - число способов разлить напитки по стаканам друзей так, чтобы все напитки полностью поместились в стаканы, взятое по модулю 1 000 000 007.

Пример

Стандартный ввод	Стандартный вывод
3	6
3	1
1 1 1	0
1 2 1	
5	
1 2 3 4 5	
1 2 3 4 5	
3	
2 2 2	
2 1 2	

Замечание

В первом тестовом наборе любой напиток помещается в любой стакан. Во втором тестовом наборе, существует лишь один способ разлить напитки. В третьем наборе ни один напиток не поместится во второй стакан, поэтому ответ будет ноль.

Решение:

Для решения данной задачи выведем комбинаторную формулу. Начнем с самого маленького стакана. Сколько различных напитков можно в него налить, не нарушая условие задачи? Ровно столько, сколько существует бутылок напитков с объемом не больше объема этого стакана. Сколько различных напитков можно налить во второй по величине стакан? Столько, сколько существует бутылок напитков с объемом не больше объема этого стакана минус один, потому что один напиток мы уже налили в первый стакан. Сколько различных...

По такой логике получаем искомую формулу $ans = cnt_1 \cdot (cnt_2 - 1) \cdot (cnt_3 - 2) \dots$. Здесь cnt_i число бутылок напитков, объем которых меньше, чем объем i -го по размеру стакана. К сожалению, если считать эту величину «в лоб», попарно сравнивая объемы бутылок и стаканов, можно получить ошибку «Превышен лимит времени исполнения». Посчитать эту величину быстрее можно, например, методом двух указателей, после сортировки обоих массивов. Таким образом получим решение, которое работает за $O(n \log n + n)$.

Отборочный этап. Первый тур (приведен один из вариантов заданий)

1. Кодирование информации. Системы счисления (2 балла)

[Без нулей]

Сколько существует целых положительных чисел таких, что их запись в восьмеричной системе счисления оканчивается на две одинаковые цифры, запись не содержит цифр 0, и сумма цифр в записи равна 8_{10} . В ответе укажите целое число.

Ответ: 43

2. Кодирование информации. Системы счисления (2 балл)

[Умножь-сложи]

Петя взял шестнадцатеричное число $0, E38_{16}$, разделил его на некоторое целое положительное число X и частное сложил с исходным числом. В результате получилось число, шестнадцатеричная запись которого содержит только цифры F после запятой. Найдите X и запишите его в ответ в десятичной системе счисления.

Ответ: 8

3. Кодирование информации. Количество информации. (3 балла)

[Буфер]

Петя конструирует систему сбора и хранения данных датчиков технологической линии. Каждую секунду необходимо записывать показания трех датчиков: K , L и M . Петя знает, что каждый датчик имеет конечный набор возможных значений

показаний. Так, для датчика К возможны 100 различных значений показаний, для датчика L – 1000, а для датчика М – 100 000. При записи данных в память используется побайтная адресация, поэтому записывается всегда целое количество байт. Сначала Петя решил использовать такой механизм записи: раз в секунду он получает значения показаний каждого датчика и независимо кодирует показание каждого датчика, используя минимальное, одинаковое для всех возможных показаний этого датчика количество бит. Затем Петя формирует последовательность из полученных двоичных кодов и сохраняет результат в память, используя минимально возможное целое количество байт. Вася предложил Пете использовать буфер: сначала в течение Т секунд записывать подряд получаемые последовательности двоичных кодов подряд в буфер, а затем сохранять получившийся блок данных в память, используя минимально возможное целое количество байт. При каком минимальном Т Петя сможет сэкономить на записи блока 20 байт по сравнению с посекундной записью в течение Т секунд? В ответе укажите целое число.

Ответ: 27

4. Кодирование информации. Количество информации (1 балл)

[Много пикселей]

Вася уже опытный инженер-программист и берется за сложные проекты. Он решил создать фотокамеру высокого разрешения. Камера делает снимки размером N на N пикселей. Пока у Васи нет мощного процессора и он не может сжимать изображения, поэтому он решил хранить для каждого пикселя файл со значениями яркости по трем каналам R, G и B, используя 8 бит для каждого значения. Таня проанализировала файлы, которые получились у Васи, и поняла, что используемые Васей сенсоры имеют не такую высокую чувствительность. Она сообщила Васе, что только в канале G можно получить такое количество уровней яркости, что для их кодирования при условии использования минимально возможного, одинакового для всех значений количества бит, потребуется ровно 8 бит. При этом в канале R количество возможных уровней яркости ровно в 2 раза меньше, чем в канале G, а в канале B – вообще в 4 раза меньше, чем в канале G. Таня предложила Васе для канала R и для канала B независимо хранить коды значений яркости используя минимально возможное, одинаковое для всех возможных яркостей в этом канале количество бит. Вася попробовал такой способ хранения и увидел, что размер файла уменьшился ровно на 96 Мбайт. Определите N и запишите в ответ целое число.

Примечание: 1 Мбайт=2²³ бит.

Ответ: 16384

5. Основы логики. Анализ логических функций (2 балла)

[Отрицания]

Для логической функции F(A,B,C,D) построена полная таблица истинности:

A	B	C	D	F(A,B,C,D)
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Известно, что функцию F(A,B,C,D) можно представить в виде следующего логического выражения:

$((\#A \rightarrow \#B) \rightarrow \#C) \rightarrow \#D$,

где вместо знаков # может стоять или отсутствовать операция логического отрицания. Расставьте операции логического отрицания так, чтобы получившаяся функция в точности соответствовала построенной таблице истинности. В ответе укажите последовательность из четырех символов «+» и «-» в порядке их следования в выражении, где «+» означает отсутствие операции отрицания перед логической переменной, а «-» – наличие операции отрицания перед логической переменной.

Ответ: +-++

6. Основы логики. Упрощение логического выражения (1 балл)

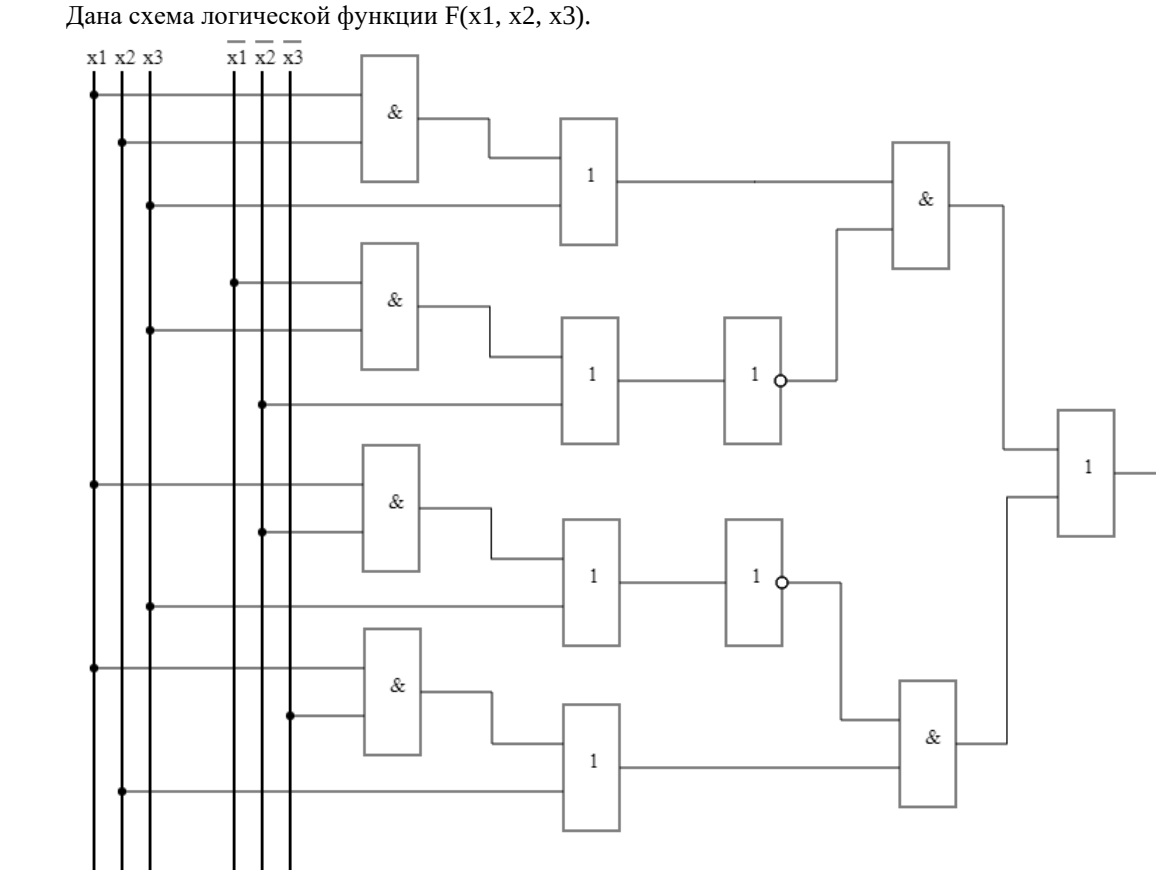
[Цепочки следствий]

Упростите логическое выражение или укажите его результат (при его однозначности). Результат упрощения может содержать только операции инверсии, конъюнкции и дизъюнкции.

$((AB \rightarrow C) \rightarrow A) \rightarrow ((BC \rightarrow D) \rightarrow B)$

Комментарий по вводу ответа: операнды вводятся большими **латинскими** буквами; логические операции обозначаются, соответственно как **not**, **and** и **or**.
 Скобки используются только для изменения порядка выполнения операций. Если порядок выполнения операций очевиден из их приоритетов – дополнительное использование скобок считается ошибкой.
 При однозначном ответе – истинный ответ обозначается как 1, а ложный как 0.

Пример записи ответа: (A or not B) and C
Ответ: not A or B | B or not A
7. Основы логики. Синтез выражения по логической схеме (2 балла)
[Логическая схема]



Сколько существует различных комбинаций значений логических переменных x1, x2 и x3, при которых на выходе будет получаться значение «ложь». В ответе укажите целое неотрицательное число.
 Примечание. На схеме использованы следующие обозначения логических операторов:

Конъюнкция	Дизъюнкция	Инверсия

Входы с горизонтальной чертой над логической переменной означают её инверсию.
Ответ: 5
8. Алгоритмизация и программирование. Формальный исполнитель (1 балл)
[Чёт-нечет]

Строки обрабатываются в соответствии со следующим алгоритмом:
 1. Заменить в строке все символы 'А' на подстроку 'СВС'
 2. Посчитать количество символов в строке. Если оно превысило 20000, то завершить исполнение алгоритма.
 3. Заменить в строке все символы 'С' на подстроку 'АВА'
 4. Подсчитать количество символов в строке. Если оно более 20000, завершить исполнение алгоритма, в противном случае перейти на шаг 1.
 Пусть исходно была строка 'АВВА'. Определите символы, которые в строке, получившейся после обработки, находятся на позициях 1000, 15001 и 30002. Позиции символов считаются слева направо, начиная с 1. В ответе укажите через пробел три символа: сначала символ на позиции 1000, затем символ на позиции 15001 и затем символ на позиции 30002.
Ответ: В С С

9. Алгоритмизация и программирование. Планирование (3 балла)

[Спираль]

Петя написал программу, которая по заданному числу N строит квадратную матрицу NxN, заполненную последовательно натуральными числами по спирали по часовой стрелке, начиная с верхнего левого угла. Например, для некоторых значений N его программа вывела такие матрицы:

N=4	N=5
1 2 3 4	1 2 3 4 5
12 13 14 5	16 17 18 19 6
11 16 15 6	15 24 25 20 7
10 9 8 7	14 23 22 21 8
	13 12 11 10 9

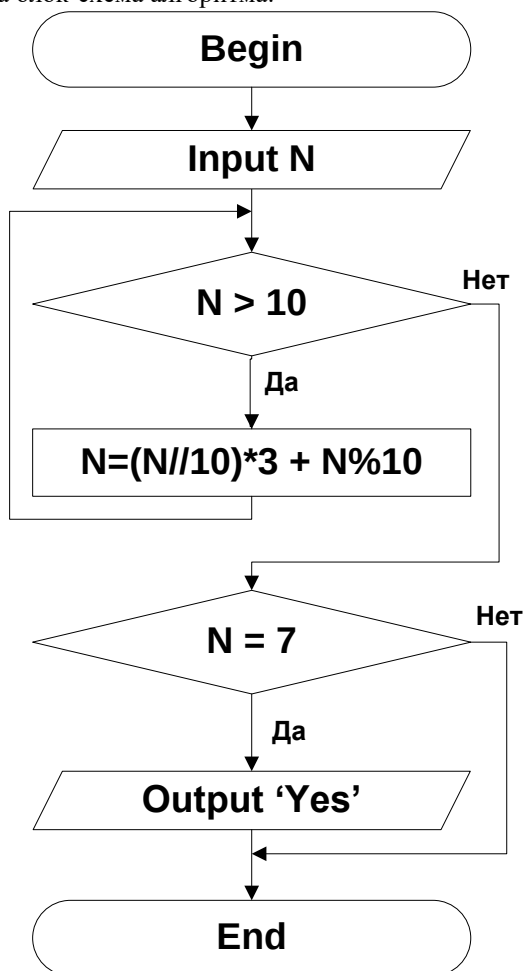
Вася предложил Пете найти такое N, при котором значение предпоследнего элемента на главной диагонали (элемента с индексами (N-1, N-1) при нумерации элементов с (1,1) от верхнего левого угла) будет равно 585. Помогите Пете и запишите в ответ найденное значение N или NULL, если такого значения N не существует.

Ответ: 99

10. Алгоритмизация и программирование. Блок-схема, обратная задача (1 балл)

[Septem]

Дана блок-схема алгоритма.



Алгоритм был выполнен для всех значений N из диапазона [1000,1200]. Для скольких значений в результате его выполнения было выведено слово 'Yes'? В ответе укажите целое число.

Примечание. Операция A//B вычисляет частное от целочисленного деления A на B. Операция A%B вычисляет остаток от целочисленного деления A на B.

Ответ: 29

Отборочный этап. Второй тур (приведен один из вариантов заданий)

1. Электронные таблицы. Адресация ячеек и вычисления (2 балла)

[25x25]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D	E
1	=СУММ(C3:AA27)				
2		0	1	=C2+2	
3		1	=ОСТАТ(C\$2;\$B3)+ОСТАТ(\$B3;C\$2)+B2		
4		=B3+2			
5					
6					

Ячейку D2 скопировали во все ячейки диапазона E2:AA2. Ячейку B4 скопировали во все ячейки диапазона B5:B27. Ячейку C3 скопировали во все ячейки диапазона C3:AA27. Затем удалили одну из ячеек диапазона C3:AA27 и получили в ячейке A1 значение 97088. Определите, какую ячейку удалили и запишите в ответ её адрес, например, C4. Если таких ячеек несколько, запишите в ответ адрес любой из них.

Ответ: P10 | J16

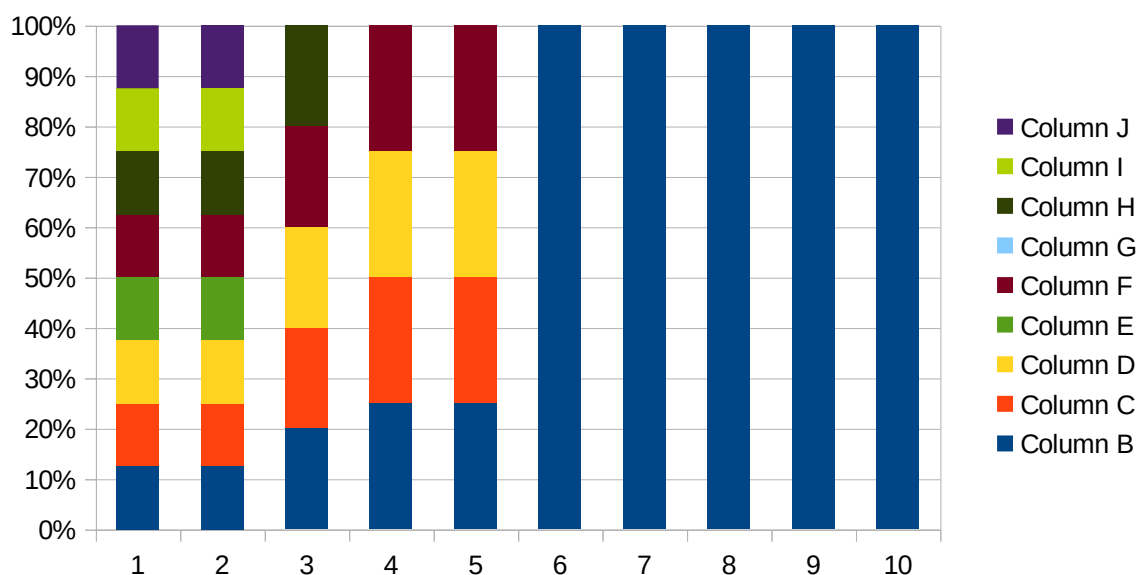
2. Электронные таблицы. Графики и диаграммы (1 балл)

[Остатки]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D	E	F	G	H	I	J	K
1	2		3	4	5	6	7	8	9	10	
2	1	=ЕСЛИ(ОСТАТ(\$A\$1;СТЕПЕНЬ(B\$1;\$A2))=0;1;0)									
3	2										
4	3										
5	4										
6	5										
7	6										
8	7										
9	8										
10	9										
11	10										
12											

Ячейку B2 скопировали во все ячейки диапазона B2:J11. В ячейку A1 поместили целое положительное число N, выделили диапазон B2:J11 и построили нормированную гистограмму с накоплением. При каком минимальном значении N могла получиться такая диаграмма:



В ответе укажите целое число.

Ответ: 6220800

3. Сортировка и фильтрация данных (3 балла)

[Большой-меньший]

Петя придумал свой алгоритм сортировки одномерного массива. Он не очень эффективный, но Пете нравится. Алгоритм состоит из следующих шагов:

1. Найти в массиве все пары элементов такие, что в них меньший по значению элемент имеет больший индекс, чем больший по значению элемент. Если таких пар нет, завершить исполнение алгоритма, иначе перейти к следующему шагу.
2. Из найденных пар элементов выбрать такую пару, разность между значениями которых будет максимальна. Если таких пар несколько, выбрать ту, у которой первый элемент имеет меньший индекс.
3. Поменять местами элементы в выбранной паре.
4. Перейти к шагу 1.

Петя взял массив [4,3,1,2] и использовал свой алгоритм. В результате он получил массив, отсортированный по возрастанию элементов. Петя обратил внимание, что шаг 3 его алгоритма в результате был исполнен 3 раза.

Тогда Петя решил взять массив с большим количеством элементов:

[10,16,6,13,11,18,8,5,9,0,2,15,4,3,7,14,12,19,1,17]

Сколько раз будет исполнен шаг 3 алгоритма Пети в этом случае?

В ответе укажите целое число.

Ответ: 38

4. Поиск и фильтрация данных (1 балл)

[Олимпиада по программированию]

В базе данных хранятся результаты олимпиады по программированию. Во время прохождения олимпиады каждый участник выполнял все задания на одном и том же языке программирования. Можно было использовать только языки программирования Python, C++, Java, Pascal. Петя сделал запросы к базе данных и получил следующие соотношения в количестве выдаваемых записей:

1. Количество записей, полученных по запросу «Участник выполнял свои задания на языке Python», ровно в 2 раза больше, чем количество записей, полученных по запросу «Участник выполнял задания на языке C++».
2. Количество записей, полученных по запросу «Участник выполнял свои задания на языке Java», ровно в 5 раз меньше, чем количество записей, полученных по запросу «Участник выполнял свои задания на языках Python или C++».
3. Количество записей, полученных по запросу «Участник выполнял свои задания на языке Pascal», составляет ровно X процентов от количества записей, полученных по запросу «Участник выполнял задания на языках C++ или Java», где X – целое число.

Как опытный участник олимпиад, Петя знает, что количество победителей не может превышать 8 процентов от количества участников, а суммарное количество победителей и призеров не может превышать 25 процентов от количества участников.

При каком минимальном X **может** быть истинным утверждение «Все победители выполняли свои задания на языке Pascal»?

В ответе укажите целое число.

Ответ: 20

5. Телекоммуникационные технологии (3 балла).

[Конкуренция за канал]

Два приложения (А и В) используют один канал связи для передачи данных. Балансировщик нагрузки на канал работает таким образом, что, если в данный момент времени передает данные только одно приложение, скорость передачи данных этого приложения составляет 96 Мбит в секунду. Если одновременно передают данные два приложения, то канал делится пополам, и скорость передачи данных каждого приложения составляет 48 Мбит в секунду. Изменения скорости могут происходить не чаще, чем раз в секунду, то есть если два приложения передавали данные одновременно, но одно из них закончило передачу до истечения очередной секунды, скорость передачи данных второго приложения не изменится до наступления новой секунды.

Приложения передают данные блоками через равные промежутки времени. Приложение А начинает передавать первый блок данных, размером в 96 Мбайт в начальный момент времени и далее передает очередной блок данных такого же размера по истечении каждой 16-ой секунды. Приложение В начинает передавать первый блок данных, размером в 60 Мбайт по истечении 4-ой секунды от начального момента времени и далее передает очередной блок такого же размера по истечении каждой 9-ой секунды от момента начала передачи своего первого блока.

Приложения не используют буферизацию, поэтому если получается ситуация, при которой одному из приложений нужно начать передачу своего очередного блока данных, но еще не закончилась передача предыдущего блока данных **этого** приложения, то происходит аварийная ситуация, передача данных обоими приложениями мгновенно останавливается и более не возобновляется. Определите, сколько блоков с данными успеет передать приложение А и сколько приложение В до возникновения аварийной ситуации. В ответе укажите через пробел два целых числа: сначала количество успешно переданных блоков приложения А и затем количество успешно переданных блоков приложения В. Если аварийной ситуации при заданных параметрах возникнуть не может, в ответе напишите NULL.

Ответ: 3 4

6. Операционные системы (2 балла)

[Shortest Job First]

Одним из алгоритмов планирования использования времени процессора в операционных системах является алгоритм Shortest Job First. Его принцип работы заключается в том, чтобы давать приоритет исполнения тому процессу, у которого наименьшее оставшееся время выполнения. Тогда "легкие" процессы будут быстрее покидать очередь к процессору и, тем самым, освобождать ресурсы вычислительного узла.

Рассмотрим модель работы планировщика. Время модели измеряется в условных тактах. Нумерация тактов идет с 1. Пусть для 5 процессов известны следующие два параметра: t_{start} - номер такта, в который данный процесс становится

готов к исполнению и cpu-burst - количество тактов процессорного времени, которые нужно предоставить процессу, чтобы он успешно исполнился. Эти данные приведены в таблице.

Процесс	t_start	cpu-burst
P1	1	13
P2	4	6
P3	7	3
P4	1	11
P5	16	4

Планировщик предоставляет возможность исполняться на процессоре очередному процессу в течение кванта непрерывного исполнения, равного 4 тактам. По истечении каждого кванта непрерывного исполнения или в случае, когда очередной процесс завершил исполнение до окончания выделенного ему кванта непрерывного исполнения, планировщик выбирает следующий процесс. Планировщик всегда выбирает тот процесс, у которого остался минимальный среди всех готовых к исполнению в данный такт процессов остаток от его cpu-burst и предоставляет ему новый квант непрерывного исполнения. Если в момент принятия решения планировщиком существует несколько процессов с одинаковым минимальным остатком cpu-burst , будет выбран процесс с меньшим порядковым номером. Если минимальный остаток cpu-burst получается у процесса, который выполнялся в предыдущем кванте непрерывного выполнения, планировщик снова выберет его для исполнения. Считается, что на само принятие решений планировщиком не тратится время и оно принимается мгновенно.

Определите, когда закончит исполняться каждый из указанных в таблице процессов.

В ответе укажите через пробел 5 целых положительных чисел – номера тактов, в которые завершат работу процессы в порядке возрастания их порядковых номеров.

Ответ: 37 10 13 20 24

7. Технологии программирования (2 балла)

[Техническое обслуживание]

Имя входного файла

стандартный ввод

Имя выходного файла

стандартный вывод

Ограничение по времени

2 секунды

Ограничение по памяти

256 мегабайт

В стране Точляндии все жители ездят на одинаковых автомобилях. Несмотря на то, что жители Точляндии выполняют техническое обслуживание точно в поставленный срок и не километром позже, многие из них постоянно забывают какие именно запчасти меняются в их автомобиле, но при этом очень любопытные и постоянно звонят на горячую линию, чтобы узнать эту информацию.

Техническое обслуживание в автомобилях Точляндии необходимо выполнять каждые 15000 километров. При этом для каждой запчасти известен срок ее службы. Так, например, масло требуется менять каждые 15000 километров, поэтому его меняют на каждом техническом обслуживании. А если срок службы запчасти 75000, ее будут менять только при пробегах 75000, 150000, 225000, 300000

Вас наняли в информационный отдел единой автомобильной компании Точляндии и просят реализовать сервис, который по пробегу автомобиля сможет определять список запчастей, которые менялись на предыдущем техническом обслуживании.

Формат входных данных

В первой строке даны два натуральных числа n и m - число сменных запчастей в автомобиле Точляндии и текущий пробег автомобиля ($1 \leq n \leq 100, 15000 < m < 1000000$). Гарантируется, что текущий пробег не является пробегом, на котором делают техническое обслуживание.

В следующих n строках даны запчасти и срок их службы в километрах. Название каждой запчасти состоит из не более чем двадцати строчных латинских букв. Гарантируется, что все названия различны. Срок службы - натуральное число кратное 15000, но не превосходящее 1000000.

Формат выходных данных

В первую строку выведите число k - число запчастей, которые менялись на предыдущем техническом обслуживании.

В следующих k строках выведите названия запчастей в лексикографическом порядке по одному названию на каждой строке.

Примеры

стандартный ввод	стандартный вывод
5 123567 interiorfilter 15000 sparkplugs 30000 oilfilter 15000 oil 15000 timingbelt 45000	4 interiorfilter oil oilfilter sparkplugs

1 100001 fullcar 990000	0
----------------------------	---

Замечание

При пробеге 123456 километров предыдущее техническое обслуживание было на пробеге 120000 километров, а следующее будет при пробеге 135000 километров.

8. Технологии программирования (4 балла)

[Взлом терминала]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

В этой задаче вам нужно взломать терминал. У терминала есть n числовых полей, чтобы его взломать необходимо, чтобы числа в полях являлись перестановкой чисел от 1 до n .

С терминалом можно делать следующие операции:

- Увеличить значение в любом числовом поле на единицу.
- Уменьшить значение в любом числовом поле на единицу.

Очевидно, что взломать терминал - это вопрос времени, но вам нужно сделать это как можно быстрее. По начальным значениям числовых полей определите минимальное число операций, необходимое, чтобы взломать терминал.

Формат входных данных

В первой строке дано одно натуральное число n - число числовых полей терминала ($1 \leq n \leq 1000$).

Во второй строке заданы n чисел $a_1, a_2, \dots, a_n$ - начальные значения числовых полей терминала ($-1000 \leq a_i \leq 1000$).

Формат выходных данных

Выведите одно число - минимальное число операций, необходимое, чтобы взломать терминал.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3	0
3 1 3 2	0
3 -1 -2 -3	12
3 -3 -2 -1	12
2 0 0	3

Задания для 7 и 8 класса

Заключительный этап (приведен один из вариантов заданий)

1. Системы счисления. 1 балл.

[Система неравенств]

Известно, что некоторое целое положительное число f может быть получено в результате вычисления следующего выражения:

$$f = 2000_N - 100_N, \text{ где } N - \text{неизвестное основание позиционной системы счисления.}$$

Дана система неравенств:

$$\begin{cases} 258_{16} \leq f < 7D0_{16} \\ 3E8_{16} \geq f > C8_{16} \end{cases}$$

Определите, для каких N эта система неравенств будет верна.

В ответе запишите через пробел все подходящие основания систем счисления N в порядке возрастания.

Ответ: 7 8

Решение:

Сначала объединим условия системы неравенств:

1. $7D0_{16} > 3E8_{16}$, следовательно $f \leq 3E8_{16}$
2. $C8_{16} < 258_{16}$, следовательно $258_{16} \leq f$
3. Получаем неравенство $258_{16} \leq f \leq 3E8_{16}$

Переведем шестнадцатеричные числа в десятичную систему счисления:

$$258_{16} = 2 \cdot 16^2 + 5 \cdot 16 + 8 = 512 + 80 + 8 = 600$$

$$3E8_{16} = 3 \cdot 16^2 + 14 \cdot 16 + 8 = 768 + 224 + 8 = 1000$$

Теперь, используя формулу развернутой записи числа для f , получаем:
 $f = 2 * N^3 - N^2$

Далее подберем такие целые положительные N , для которых выполняется неравенство $258_{16} \leq 2 * N^3 - N^2 \leq 3E8_{16}$. Для данного неравенства подходят только два целых значения N : 7 и 8.

Можно предложить также программное решение:

```
# Переберем все основания систем счисления до достаточно большого числа
for n in range(2, 37):
    # Запишем число f в многочленной форме
    f = 2 * n ** 3 - n ** 2
    # Проведем проверку из условия задачи
    if int('258', 16) <= f < int('7d0', 16) and int('c8', 16) < f <= int('3e8', 16):
        print(n)
```

2. Кодирование информации. 1 балл.

[Новый шифр]

Петя придумал новый шифр для текстов, записанных с использованием 32-х символов русского алфавита (без буквы ё):

1. Для шифрования используется таблица из черных и белых квадратов размером 4*8, например, такая:

■	□	□	□	□	□	□	□
□	■	□	□	□	□	□	□
□	□	■	□	□	□	□	□
□	□	□	■	□	□	□	□

2. Каждой букве алфавита сопоставляется число от 1 до 32. Так же каждой клетке таблицы сопоставляется число от 1 до 32 (нумерация идёт слева-направо сверху-вниз). Например, черная клетка во второй строке будет иметь номер 10.
3. Строится новый алфавит для шифрования следующим образом. Просматриваем ячейки таблицы в порядке возрастания их номеров. Если встречается ячейка черного цвета, то буква, соответствующая порядковому номеру ячейки, перемещается в новом алфавите со своей позиции в конец.
4. В результате получается новый алфавит, который записывается под исходным алфавитом. Для приведенного примера таблицы получится следующий результат:

а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
б	в	г	д	е	ж	з	и	к	л	м	н	о	п	р	с	у	ф	х	ц	ч	ш	щ	ъ	ь	э	ю	я	а	й	т	ы

Для шифрования каждая буквы исходного текста меняется на букву, находящуюся под ней.

Тогда, используя таблицу из примера можно зашифровать фразу «шифр удался», и зашифрованный текст будет выглядеть следующим образом: «ъкчу цебнфы».

Петя использовал таблицу, приведенную ниже, чтобы зашифровать некоторое слово:

□	■	□	□	■	□	■	□
■	■	□	□	□	■	□	□
□	□	■	■	□	□	■	■
□	■	□	□	■	□	■	□

У него получилось слово «ваывазшэ». Каким было исходное слово до шифрования?

Ответ: барбадос

Решение:

Исходя из таблицы, нужно перемещать в конец символы со следующими номерами: 2, 5, 7, 9, 10, 14, 19, 20, 23, 24, 26, 29 и 31.

Составив алфавит для шифрования, перемещая соответствующие символы, получим:

а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
а	в	г	е	з	к	л	м	о	п	р	с	ф	х	ш	ъ	ы	э	я	б	д	ж	и	й	н	т	у	ц	ч	щ	ь	ю

Поскольку для шифрования каждая буква слова заменялась с верхней строчки на букву под ней, то для расшифровки нам нужно брать букву из нижней строчки и заменять её на букву над ней.

Получаем из «ваывазшэ» слово «барбадос», это и есть ответ на задачу.

Можно предложить также программное решение:

```
# Введем эталонный русский алфавит без буквы ё
alph = 'абвгдежзийклмнопрстуфхцчшщъыьэюя'
# Организуем два списка: tmp_start - для букв алфавита, которые не перемещаются в конец
# tmp_end - для букв алфавита перемещенных в конец
tmp_start = []
tmp_end = []
```

```

# Закодируем картинку из условия в виде списка нулей (белых клеток) и единиц (четных
клеток)
code = [0, 1, 0, 0, 1, 0, 1, 0,
        1, 1, 0, 0, 0, 1, 0, 0,
        0, 0, 1, 1, 0, 0, 1, 1,
        0, 1, 0, 0, 1, 0, 1, 0]

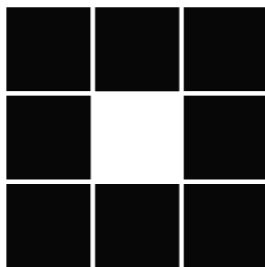
# Смоделируем создание алфавита для кодирования, описанное в условии
tmp = []
for i in range(32):
    if code[i] == 1:
        tmp_end.append(alph[i])
    else:
        tmp_start.append(alph[i])
alph_new = ''.join(tmp_start + tmp_end)
# Слово, которое нужно раскодировать
word = 'ваывазшэ'
# В новом алфавите находим место (индекс) из word и берем букву, стоящую на этой позиции
в старом алфавите
res = []
for x in word:
    res.append(alph[alph_new.find(x)])
# Выводим результат
print(''.join(res))

```

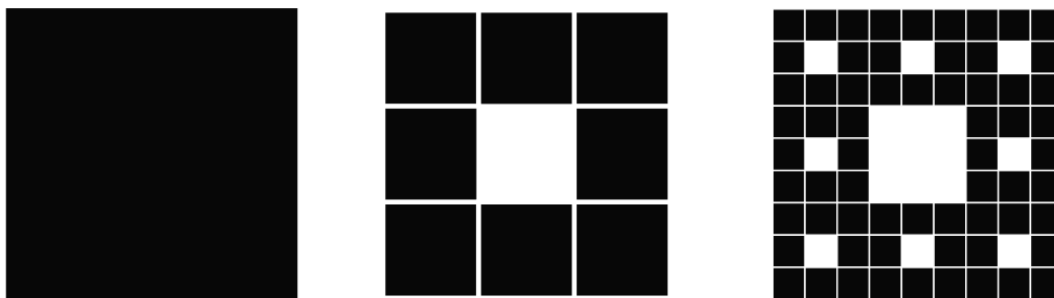
3. Кодирование графической информации. 2 балла.

[Ковёр Серпинского]

Петя и Вася узнали новый для себя фрактал – ковёр Серпинского, это фигура, которая меняется на каждом шаге. За один шаг все черные квадраты заменяются на 8 черных квадратов и один белый квадрат меньшего размера, расположенные следующим образом:



На картинке изображены нулевой, первый и второй шаги построения ковра Серпинского.



Сколько будет квадратов самого маленького размера (суммарно и черных, и белых) на 5-ом шаге? Например, на втором шаге 72 самых маленьких квадрата (64 черных и 8 белых). В ответе укажите целое число.

Ответ: 36864

Решение:

Согласно условию изменения фигуры «Ковёр Серпинского», каждый черный квадрат заменяется на 8 черных и 1 белый. На нулевом шаге мы имеем один черный квадрат.

На каждом следующем шаге самых маленьких квадратов будет:

Белых: $w = b'$

Черных: $b = b' \cdot 8$

Суммарно: $w + b = b' \cdot 9$, где

w – количество самых маленьких белых квадратов на данном шаге,
 b – количество самых маленьких черных квадратов на данном шаге,
 b' – количество черных квадратов на предыдущем шаге.

Количество черных квадратов на каждом шаге легко узнать. Поскольку каждый черный квадрат заменяется на 8 черных квадратов меньшего размера, то:

на шаге 0 – 1 черный квадрат,
на шаге 1 – 8 черных квадратов,
на шаге 2 – 64 черных квадрата,

...

на шаге $n - 8^n$ черных квадратов.

Таким образом, найдём формулу для нахождения общего числа самых маленьких квадратов на шаге n :

$$ANS(n) = 8^{n-1} * 9$$

Подставим номер шага из задачи

$$ANS(5) = 8^{5-1} * 9 = 8^4 * 9 = 36864, \text{ это и есть ответ на задачу.}$$

4. Алгебра логики. 3 балла.

[Полином Жегалкина]

Петя узнал о новом представлении логических функций – полином Жегалкина. В этом представлении функция задается выражением, в котором используются не привычные булевы операции (НЕ, И, ИЛИ), а операции «И($\wedge$)» и «Исключающее ИЛИ($\oplus$)» и константа «истина (1)».

Для двух указанных операций представлены таблицы истинности ниже:

И($\wedge$)			Исключающее ИЛИ($\oplus$)		
A	B	$A \wedge B$	A	B	$A \oplus B$
0	0	0	0	0	0
0	1	0	0	1	1
1	0	0	1	0	1
1	1	1	1	1	0

Примечание: В полиноме Жегалкина «Исключающее ИЛИ($\oplus$)» имеет наименьший приоритет.

Пример: Дана функция в полиноме Жегалкина: $1 \oplus A \oplus A \wedge B$.

Таблица истинности для данной функции:

A	B	$A \wedge B$	$1 \oplus A$	$1 \oplus A \oplus A \wedge B$
0	0	0	1	1
0	1	0	1	1
1	0	0	0	0
1	1	1	0	1

Как видно, при построении таблицы истинности функции сначала построили столбец для операции «И», как имеющей наивысший приоритет ($A \wedge B$). Затем для первой операции «Исключающее ИЛИ» ($1 \oplus A$) и затем для операции «Исключающее ИЛИ» между результатами в построенных перед этим двух столбцах.

Для данной функции будет три набора аргументов с истинным значением функции.

Вам даны несколько функций, представленных в полиноме Жегалкина:

- $C \oplus B \oplus B \wedge C \oplus A \oplus A \wedge B \wedge C$
- $1 \oplus C \oplus B \oplus A \oplus A \wedge C \oplus A \wedge B$
- $1 \oplus C \oplus B \oplus A$
- $1 \oplus C \oplus A \wedge B \wedge C$

Помогите Пете найти функции, для каждой из которых будет не менее 5 различных наборов аргументов, при которых функция даёт истину.

В ответе укажите номера подходящих функций через пробел в порядке возрастания.

Ответ: 1 4

Решение:

Построим для каждой функции таблицу истинности и узнаем количество наборов аргументов, при которых функция даёт истину.

При построении можно не расписывать столбцы для каждой операции «исключающее ИЛИ», поскольку данная операция является также «сложением по модулю 2», иначе говоря, «исключающее ИЛИ» для нескольких переменных даёт истину тогда и только тогда, когда количество слагаемых, равных истине – нечетно.

Так же для операции «исключающее ИЛИ» не важен порядок, а значит слагаемые можно менять местами для удобства.

- $C \oplus B \oplus B \wedge C \oplus A \oplus A \wedge B \wedge C$

A	B	C	$B \wedge C$	$A \wedge B \wedge C$	$A \oplus B \oplus C \oplus B \wedge C \oplus A \wedge B \wedge C$
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	1	0	1
1	0	0	0	0	1
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	1	1	1

Для данной функции 5 наборов аргументов дают истину.

- $1 \oplus C \oplus B \oplus A \oplus A \wedge C \oplus A \wedge B$

A	B	C	$A \wedge B$	$A \wedge C$	$1 \oplus A \oplus B \oplus C \oplus A \wedge B \oplus A \wedge C$
---	---	---	--------------	--------------	--

0	0	0	0	0	1
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	1	0	0
1	1	1	1	1	0

Для данной функции 2 набора аргументов дают истину.

3. $1 \oplus C \oplus B \oplus A$

A	B	C	$1 \oplus A \oplus B \oplus C$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Для данной функции 4 набора аргументов дают истину.

4. $1 \oplus C \oplus A \wedge B \wedge C$

A	B	C	$A \wedge B \wedge C$	$1 \oplus C \oplus A \wedge B \wedge C$
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	1
1	1	1	1	1

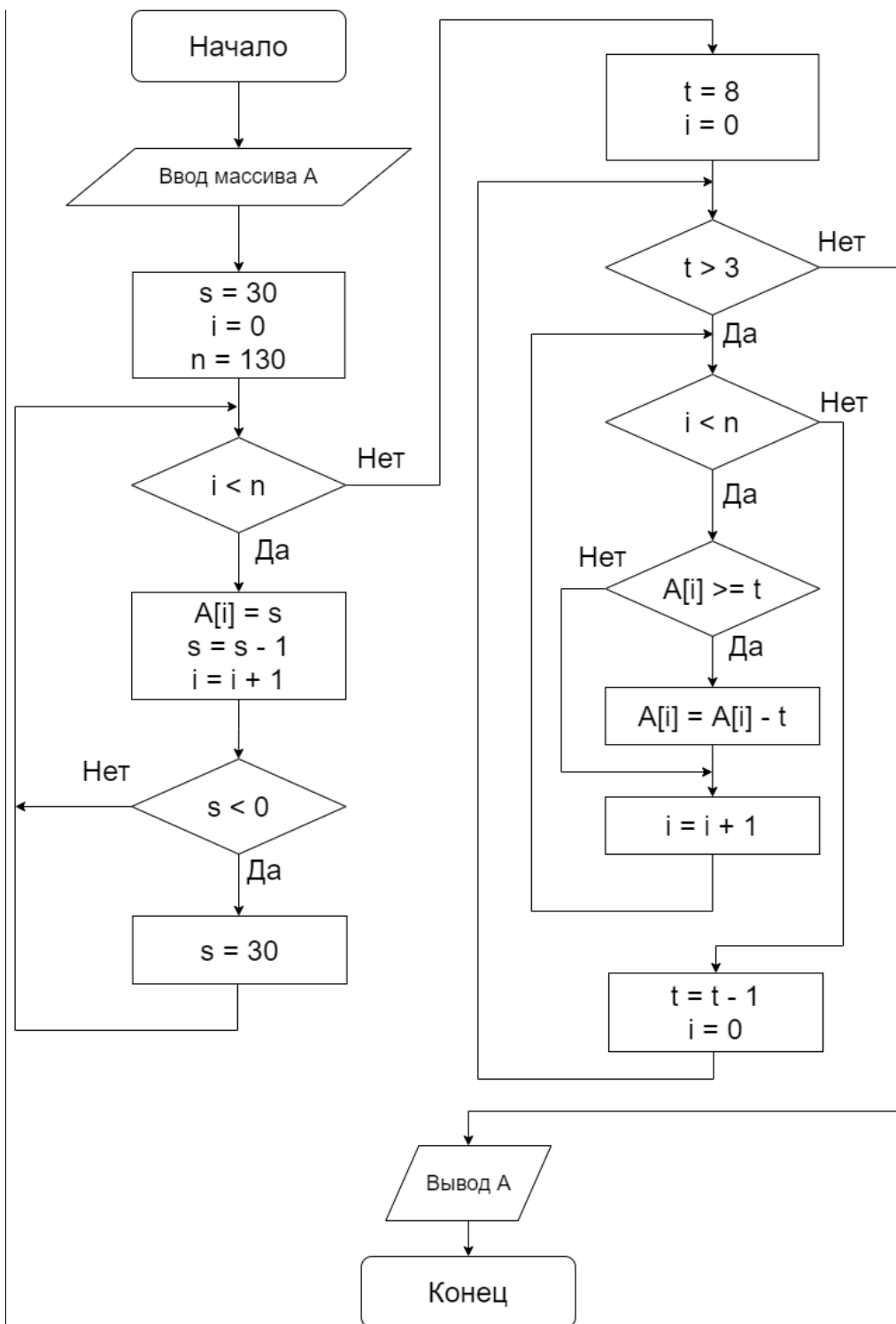
Для данной функции 5 наборов аргументов дают истину.

Таким образом, только функции 1 и 4 имеют не менее 5 наборов аргументов, при которых функция имеет истинное значение.

5. Анализ алгоритма, заданного в виде блок-схемы. 3 балла.

[Обработка массива]

Вам дана блок-схема алгоритма:



Известно, что на вход был подан массив А из 130 элементов, равных 0. Нумерация элементов массива начинается с нуля. Укажите, сколько нулей в массиве А будет после окончания работы описанного выше алгоритма?
В ответе укажите одно число – искомое количество нулей.

Ответ: 67

Решение:

Проанализируем левую часть блок-схемы. В данной части алгоритма массив А заполняется числами, посмотрим несколько чисел массива после выполнения этой части алгоритма:

30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Очевидно, что дальше последовательность будет содержать повторения приведенного фрагмента. Причем, поскольку всего чисел в последовательности должно быть 130, то повторений приведенного фрагмента будет 4 и еще шесть первых чисел из пятого повторения приведенного фрагмента:

30 29 28 27 26 25

Теперь приступим к правой части алгоритма. В ней берется некое число t, меняющееся от 8 до 4 включительно. Для каждого t алгоритм проходит по всему массиву и вычитает из каждого числа t, если $t \leq A[i]$. Составим таблицу для каждого t для первого фрагмента из 31 числа:

Фрагмент (первые 31 элемент)	t	Количество нулей
30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0	8	2
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 6 5 4 3 2 1 0 0 6 5 4 3 2 1 0	7	4
9 8 7 6 5 4 3 2 1 0 5 4 3 2 1 0 0 5 4 3 2 1 0 0 0 5 4 3 2 1 0	6	7
4 3 2 1 0 4 3 2 1 0 0 4 3 2 1 0 0 0 4 3 2 1 0 0 0 0 4 3 2 1 0	5	11
0 3 2 1 0 0 3 2 1 0 0 0 3 2 1 0 0 0 0 3 2 1 0 0 0 0 0 3 2 1 0	4	16

Следовательно, во всех остальных фрагментах произойдёт то же самое, поскольку они одинаковы.

Вычислим общее число нулей в последовательности по завершении выполнения алгоритма (необходимо помнить, что в последовательности 4 полных фрагмента и еще 6 чисел):

$$16 * 4 + 3 = 67$$

Это и есть ответ на задачу.

Можно предложить также программное решение:

Запишем блок-схему в виде алгоритма на Python

```
a = [0] * 130
```

```
s = 30
```

```
n = 130
```

```
for i in range(n):
```

```
    a[i] = s
```

```
    s = s - 1
```

```
    if s < 0:
```

```
        s = 30
```

```
t = 8
```

```
while t > 3:
```

```
    for i in range(n):
```

```
        if a[i] >= t:
```

```
            a[i] -= t
```

```
        t -= 1
```

```
# Найдем количество нулей в списке
```

```
print(a.count(0))
```

6. Анализ алгоритма, заданного в виде программного кода. 2 балла

[Три символа]

Петя придумал алгоритм, который рисует картинки с использованием трех символов: *, + и #.

алг

нач цел x, y

 x := 0

 нц пока x < 10

 y := 0

 нц пока y < 10

 если F(x, y) >= 0 и G(x, y) >= 0

 то

 Вывод '*'

 все

 если G(x, y) < 0

 то

 Вывод '+'

 все

 если F(x, y) < 0 и G(x, y) >= 0

 то

 Вывод '#'

 все

 y := y + 1

 кц

 Вывод '\n'

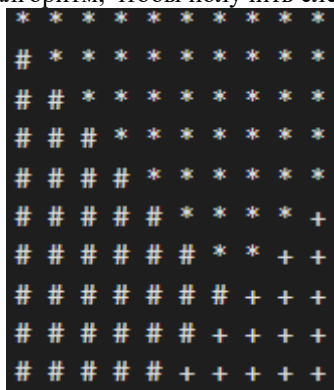
 x := x + 1

кц

кон

Во время исполнения алгоритма происходит вычисление значений двух функций: F(x,y) и G(x,y). Известно, что каждая из них представляет собой выражение вида (a*x + b*y + c), где a, b и c могут быть различными числами.

Петя дал Васе список выражений, которые можно подставить вместо $F(x, y)$ и $G(x, y)$, помогите Васе выбрать два выражения, которые необходимо подставить в алгоритм, чтобы получить следующую картинку:



Список выражений, который Петя дал Васе.

1. $-10 * x + 10 * y + 0$
2. $-4 * x + 5 * y + 0$
3. $-7 * x - 7 * y + 91$
4. $-6 * x + 6 * y + 20$
5. $x + y - 10$

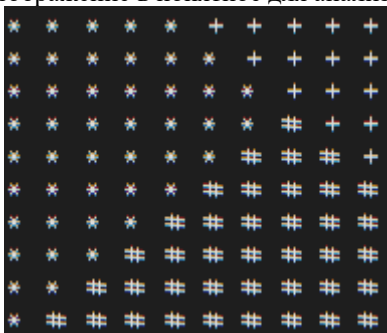
В ответе укажите два числа - сначала номер выражения, которое скрывается за $F(x, y)$, а потом за $G(x, y)$ без запятых и пробелов.

Примечание: Вывод '\n' – вывод символа переноса строки.

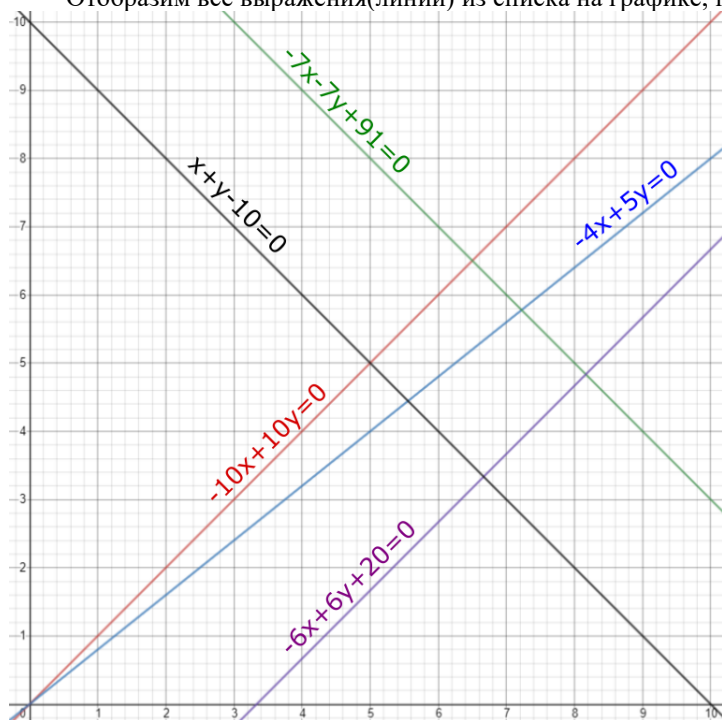
Ответ: 13

Решение:

Несложно заметить, что получившая картинка – ни что иное, как 1 четверть декартовой системы координат, повернутая на 90 градусов по часовой стрелке, а выражения $F(x, y)$ и $G(x, y)$ задают некоторые прямые в этой системе координат, разделяющие плоскость на 4 части. Вернем изображение в понятное для анализа положение:



Отобразим все выражения(линии) из списка на графике, получаем:



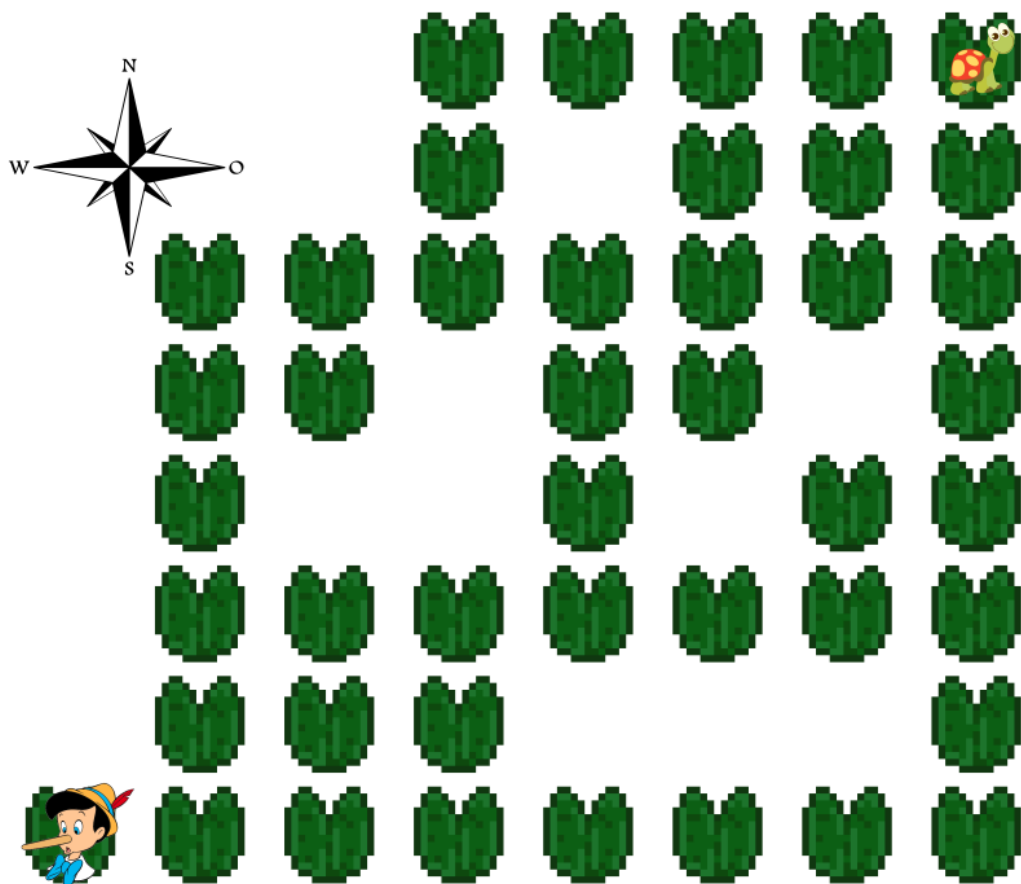
Далее попробуем сопоставить каждой целой координатной точке символ из рисунка, получаем:


```
# Сравниваем полученную строку с эталоном
if t == etalon:
    print(f, g)
```

7. Информационное моделирование. 2 балла.

[И снова Тортилла]

Когда Буратино пришел к мудрой черепахе Тортилле, она отправилась по кувшинкам за Золотым Ключиком. Но когда собралась идти обратно, заметила, что те кувшинки, по которым она шла сюда, переместились и обратный путь теперь совсем другой и очень витиеватый. Тортилла начинает движение в правом верхнем углу и может идти только на юг и на запад. Помогите Тортилле узнать, сколько у неё вариантов составить маршрут до нижней левой кувшинки. В ответе укажите целое число.



Ответ: 99

Решение:

Чтобы решить данную задачу нужно посчитать количество вариантов дойти от старта до каждой кувшинки как сумму количества вариантов дойти до кувшинки сверху и кувшинки справа от текущей кувшинки (если такие имеются). До первой кувшинки считаем, что есть только 1 вариант дойти. Далее будем считать ответ для каждой кувшинки, складывая количество вариантов дойти до тех кувшинок, из которых в неё можно попасть:

			1	1	1	1	1
			1		3	2	1
7	7	7	6	6	3	1	
14	7		12	6		1	
14			12		1	1	
28	14	14	14	2	2	1	
56	28	14				1	
99	99	43	15	1	1	1	1

В левом нижнем углу и есть ответ на поставленную задачу.

8. Поиск и фильтрация данных. 1 балл.

[База студентов]

Петя и Вася получили доступ к базе данных, содержащей результаты опроса студентов Университета. В базе данных для каждого студента хранятся следующие данные: пол студента (мужской, женский), выбранный формат обучения (очное, дистанционное), а также его уровень образования (бакалавриат, специалитет или магистратура). По каждому из указанных трех параметров у каждого студента выбрано ровно одно значение.

Известно количество записей, полученных в ответ на ряд запросов к этой базе:

1. Пол студента = «женский» и формат обучения = («очное» или «дистанционное») и уровень образования = «специалитет» - 14 записей
2. Пол студента = «женский» и формат обучения = «дистанционное» и уровень образования = «специалитет» - 0 записей
3. Пол студента = «мужской» и формат обучения = «дистанционное» и уровень образования = («бакалавриат» или «специалитет» или «магистратура») – 42 записи
4. Пол студента = «мужской» и формат обучения = («очное» или «дистанционное») и уровень образования = («бакалавриат» или «специалитет» или «магистратура») – 74 записи
5. Пол студента = («мужской» или «женский») и формат обучения = «очное» и уровень образования = «специалитет» - 21 записей
6. Пол студента = («мужской» или «женский») и формат обучения = «очное» и уровень образования = («бакалавриат» или «магистратура») – 66 записей
7. Пол студента = «женский» и формат обучения = («очное» или «дистанционное») и уровень образования = («бакалавриат» или «магистратура») – 80 записей

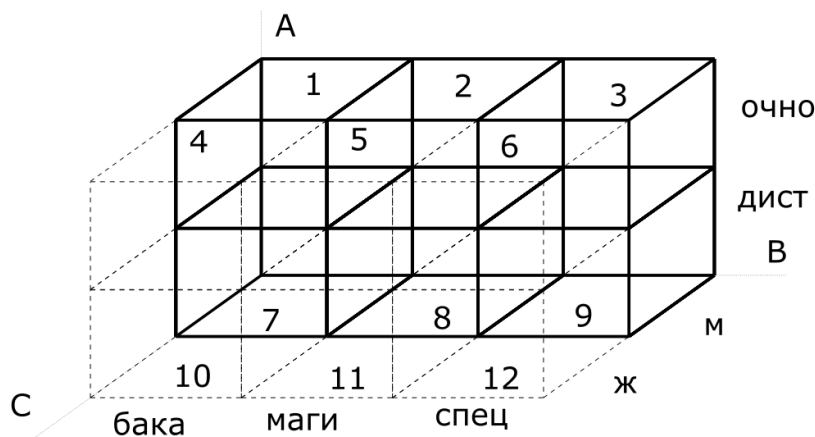
Определите, сколько студентов выбрали дистанционный формат обучения?

В ответе укажите одно число.

Ответ: 81

Решение:

Как видно из запросов, в них участвуют значения трех полей, следовательно, все записи можно представить как области пространства значений полей с заданной на нем декартовой системой координат, где в качестве осей будут выступать поля: «Пол студента», «Формат обучения» и «Уровень образования». Как видно из условия, поля «Пол студента» и «Формат обучения» имеют два варианта значения, а «Уровень образования» три варианта значений. Следовательно, любая запись в базе данных будет принадлежать одной из 12 областей:



Тогда первому запросу соответствуют области 6,12

Второму запросу соответствует область 12

Третьему запросу соответствуют области 7,8,9

Четвертому запросу соответствуют области 1,2,3,7,8,9

Пятому запросу соответствует область 3,6

Шестому запросу соответствуют области 1,2,4,5

Седьмому запросу соответствуют области 4,5,10,11

Для определения количества студентов на дистанционном формате обучения необходимо найти, сколько запросов соответствуют областям 7,8,9,10,11,12.

- 1) Из первых двух запросов легко найти, что в области 12 – 0 записей, а в области 6 – 14 записей
- 2) Из первого пункта и пятого запроса видно, что в области 3 – 7 записей
- 3) Из второго пункта и третьего запроса видно, что суммарно в областях 1,2 – 25 записей
- 4) Из третьего пункта и шестого запроса видно, что суммарно в областях 4,5 – 41 запись
- 5) Из четвертого пункта и седьмого запроса видно, что суммарно в областях 10,11 – 39 записей
- 6) Из первого и пятого пунктов, а также третьего запроса видно, что в областях 7,8,9,10,11,12 суммарно $42+39+0=81$ запись

Это и будет правильным ответом.

9. Обработка информации в электронных таблицах. 2 балла

[Запутанные формулы]

Петя показал Васе фрагмент таблицы:

	A	B	C
1	=B1*B2	7	=A2*B4 / 11
2	=B2+10	12	=B3*B5
3	=C3	10	=A1+C2-8
4	=-(C4 + B4*(B3-B1)*(B3-B1))		=C1+B4*(B2-B1)
5	=C5*1/(B5-B1)	15	=B4*B4*(B2-B3)
6			
7		=СУММ(A3:A5)*СУММ(A3:A5)	
8			

Петя также задал Васе вопрос, какое целое положительное число должно быть в ячейке B4, чтобы после выключения режима отображения формул в ячейке B7 оказалось число **900**?

Помогите Васе найти ответ на этот вопрос. В ответе укажите одно число – искомый ответ на задачу.

Ответ: 32

Решение:

Поскольку в ячейке B7 формула $(A3+A4+A5)^2$ после выключения режима отображения формул оказалась равна 900, тогда $A3+A4+A5 = \pm 30$.

Следующим шагом найдём несколько значений ячеек, которые уже известны:

	A	B	C
1	=7*12=84	7	=A2*B4/11
2	=12+10=22	12	=10*15=150
3	=226	10	=84+150-8=226
4	=-(C4+B4*(B3-B1)*(B3-B1)) x		=C1+B4*(B2-B1)
5	=C5*1/(B5-B1)	15	=B4*B4*(B2-B3)

В оставшихся формулах заменим ячейку B4 на x, а остальные известные ячейки на числа и уже посчитанные формулы с x:

	A	B	C
1	84	7	=22*x/11=2*x
2	22	12	150
3	226	10	226
4	=-(7*x+x*(10-7)*(10-7))=-7*x-9*x=-16*x x		=2*x+x*(12-7)=2*x+x*5=7*x
5	=2*x*x*1/(15-7)=2*x*x/8=x*x/4	15	=x*x*(12-10)=2*x*x

Из формулы $A3+A4+A5 = \pm 30$ и известных формул в ячейках A3, A4 и A5 получим равенство:

$$\frac{x^2}{4} - 16 * x + 226 = \pm 30$$

Теперь есть два пути решения.

Первый, аналитический – решив два квадратных уравнения (для 30 и -30 в правой части), получить три возможных значения x: $32+4*\sqrt{15}$; $32-4*\sqrt{15}$; 32

Из данных ответов нам подходит только 32, поскольку по условию требуется целое положительное число.

Второй путь – перебрать значения в ячейке B4 вручную или используя макрос, например, такой:

```
f = 0
i = 1
While f = 0
    Cells(4, 2) = i
    If Cells(7, 2) = 900 Then
        Cells(1, 5) = i
        f = 1
    End If
    i = i + 1
Wend
```

10. Информационное моделирование. 1 балл

[В путь за диаграммами]

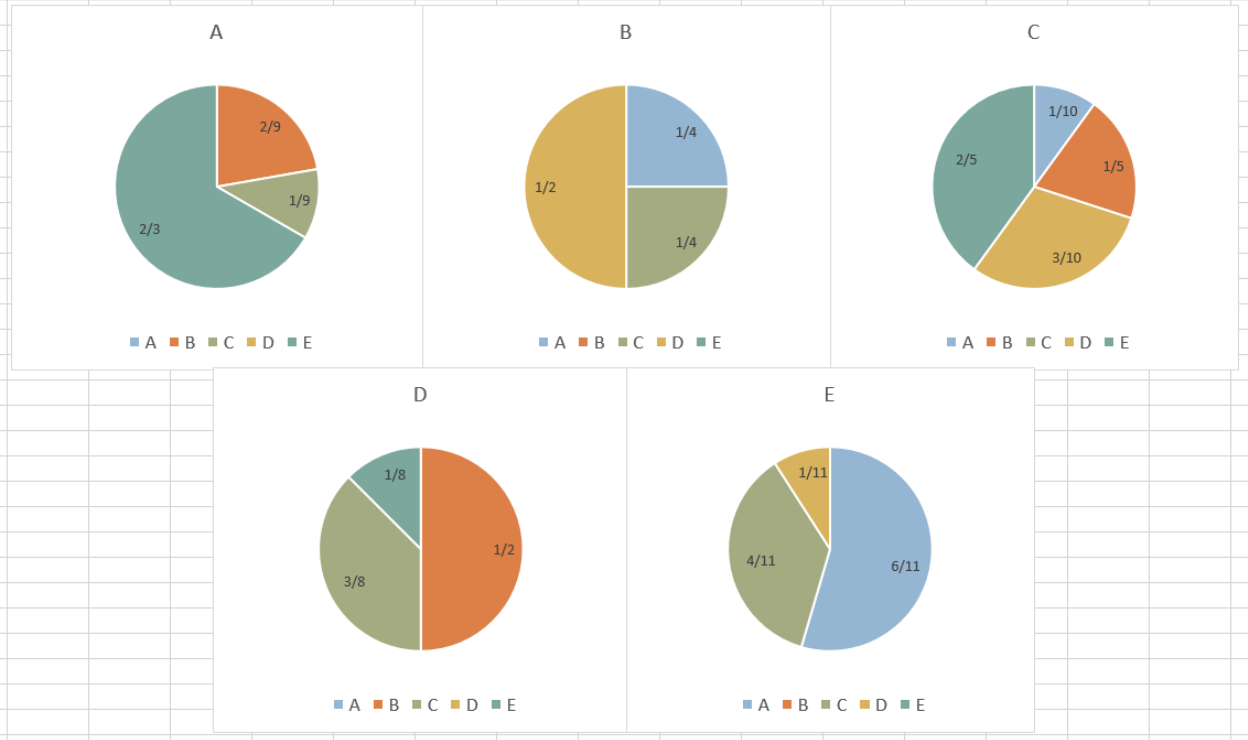
Петя любит изучать дорожные карты. Он рассматривает дорожную сеть, соединяющую 5 населенных пунктов (A, B, C, D, E) с указанием длин дорог, соединяющих два населенных пункта. Петя решил описать эту сеть в виде таблицы:

	A	B	C	D	E
A					
B					

C					
D					
E					

В такой таблице на пересечении некоторой строки и столбца записывается длина пути, если дорога между двумя населенными пунктами есть, в противном случае – пусто. Если есть дорога, соединяющая некоторый пункт X с пунктом Y, то есть и дорога, соединяющая пункт Y с пунктом X, той же длины. Ни один пункт не соединен дорогой с самим собой (главная диагональ таблицы всегда пустая), то есть нет дороги из A в A, из B в B и т.д.

Вася очень любит круговые диаграммы и знает, что по этой таблице можно составить 5 круговых диаграмм (по одной для каждой строки). Вася решил узнать, как хорошо Петя разобрался в дорожной карте и показал Пете получившиеся диаграммы:



A также Вася подсказал Пете, что дорога, соединяющая пункты A и B, имеет длину 14. Вася спросил Петю, какая минимальная длина пути между населенными пунктами A и E?

Помогите Пете, в ответе укажите одно целое число – найденную длину кратчайшего пути.

Ответ: 35

Решение:

Заметим, что раз для каждой пары пунктов X и Y есть ровно одна дорога в обе стороны, то таблица будет симметрична относительно главной диагонали (строка A равна строке-столбцу A, строка B равна строке-столбцу B и т.д.).

Нам известна из условия длина дороги между пунктами A и B, занесём её в таблицу:

	A	B	C	D	E
A		14			
B	14				
C					
D					
E					

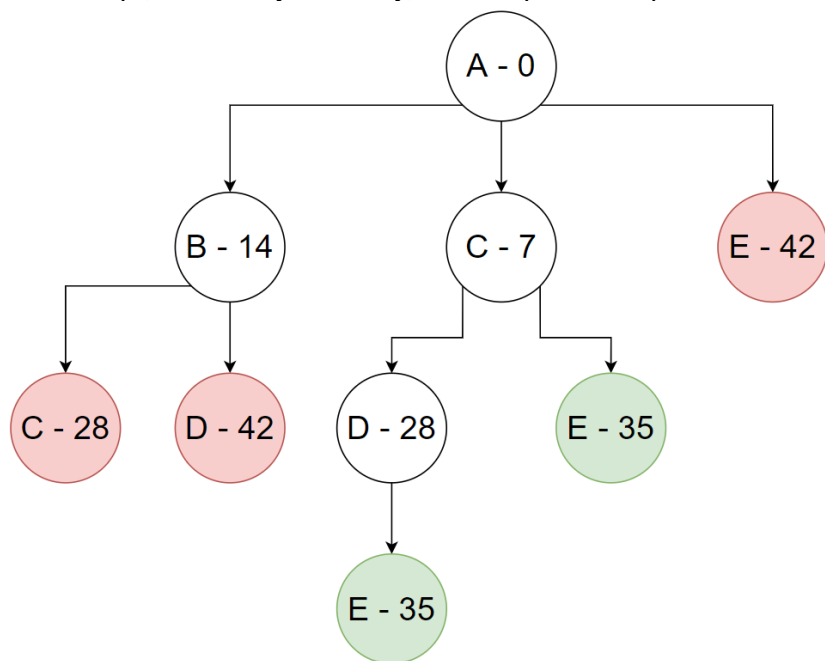
Далее, зная хотя бы одну длину дороги в строке, можно по диаграмме узнать протяженность остальных, составляя пропорции. Заполним строки и столбцы A и B:

	A	B	C	D	E
A		14	7		42
B	14		14	28	
C	7	14			
D		28			
E	42				

Теперь осталось только заполнить строки и столбцы C и D:

	A	B	C	D	E
A		14	7		42
B	14		14	28	
C	7	14		21	28
D		28	21		7
E	42		28	7	

Теперь, имея полную таблицу, найдём кратчайшее расстояние от А до Е:



Пояснение: красным выделены неэффективные маршруты. Маршрут от А до С длиной 28 менее выгоден, чем маршрут длины 7, поэтому мы его не рассматриваем. Маршрут от А до D длиной 42 не выгоден, так как прямой маршрут от А до Е имеет ту же длину. Ну и наконец прямой маршрут от А до Е длиной 42 менее выгоден, чем найденные два маршрута длиной 35.

Это и есть ответ на поставленную задачу – 35.

Отборочный этап. Первый тур (приведен один из вариантов заданий)

1. Теоретические основы информатики, одиночный выбор. (1 балл)

[Теория алгоритмов]

Какой из данных терминов является свойством алгоритмов?

1. Массовость
2. Доступность
3. Конфиденциальность
4. Целостность

В ответе укажите только число, соответствующее правильному ответу.

Ответ: 1

2. Теоретические основы информатики, множественный выбор. (1 балл)

[Архитектура сетей]

Какой(ие) из представленных сетевых протоколов являются протоколами прикладного уровня?

1. HTTPS
2. ESP
3. PPTP
4. FTP
5. NCP

Ответ: 1 4

3. Комбинаторика. (1 балл)

[D20 куб]

Как-то раз группа друзей собрались поиграть в настольную приключенческую игру Demons&Dwarfs. В этой игре, чтобы сделать какое-либо действие, нужно бросать двадцатигранные кубики. В сложной ситуации одному из мальчиков нужно было воспользоваться мощным артефактом, от которого зависел успех всей команды. Мальчик кидает два разноцветных двадцатигранных кубика: красный и синий. У каждого кубика грани содержат неповторяющиеся числа от 1 до 20. Для получения артефакта сумма на выпавших гранях должна равняться 33. Помогите ему узнать, сколько вариантов бросков кубиков такие, что на кубиках суммарно выпадет 33? Варианты, отличающиеся цветами кубиков, например, вариант 15 на синем кубике и 18 на красном и вариант 18 на синем кубике и 15 на красном, считаются различными.

В ответе укажите целое число.

Ответ: 8

4. Системы счисления. (2 балла)

[Волшебное число... или все же слово?]

Путешественник забрел в волшебную город в поисках сокровищ, славы и приключений. На входе его встретил страж и сказал, что не пропустит, пока тот не скажет кодовое слово. Чтобы узнать это кодовое слово, нужно было разгадать загадку, которая выбита в стенах города:

$$201445_6 = \text{_____}_{36}$$

Путешественник догадался, что нижние индексы означают основания систем счисления, а также он знал, что при основаниях систем счисления, больших 10, для цифр больше 9 используются последовательно заглавные буквы латинского алфавита.

Помогите путешественнику разгадать загадку, укажите в ответе кодовое слово заглавными латинскими буквами.

Ответ: CAT

5. Кодирование информации, звук. (3 балла)

[Меломан Арсений]

Арсений и его друг Паша очень любят группу AwesomePony. Как только вышел новый альбом, Паша тут же скачал его в идеальном качестве в одноканальной записи без сжатия и захотел поделиться с Арсением. Но у Арсения было свободно только 72 MB на флешке, и никакие другие файлы удалять нельзя. Тогда Паша предложил перекодировать каждую песню альбома с одинаковым разрешением, равным 16 bit, и некоторой одинаковой частотой дискретизации. Ниже представлена таблица с продолжительностью каждой песни альбома:

Название песни	Продолжительность (минуты:секунды)
Princess Fanta	2:34
Happy elephants	2:26
Marshmallow generator	3:47
Paradise	2:10
Mushrooms vs. stars	3:13

Помогите ребятам выбрать такую наибольшую целую частоту дискретизации, при которой все песни альбома поместятся на флешку. Известно, что при записи каждой песни в памяти сохраняются только значения уровней сигнала с определенной частотой дискретизации и разрешением, и не хранится никаких дополнительных данных, при этом под каждую песню выделяется целое число блоков размером в 1024 Бита.

В ответе укажите значение частоты дискретизации в Гц.

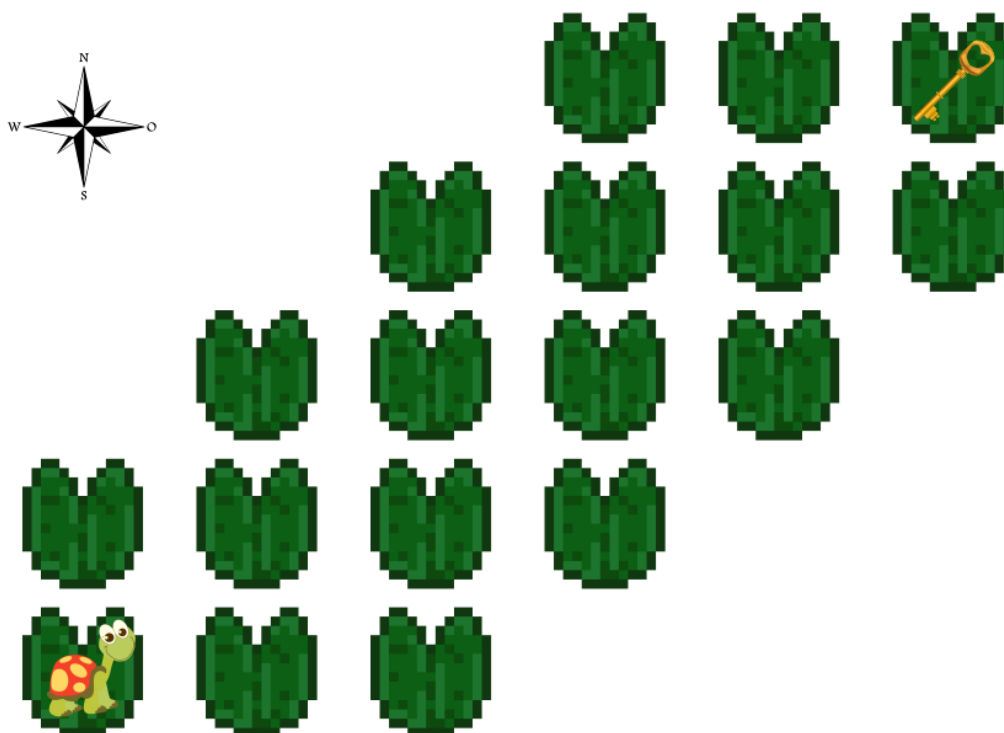
Ответ: 44408

6. Алгоритмизация и программирование. (2 балла)

[Тортилла и золотой ключик]

Буратино пришел к черепахе Тортилле за золотым ключиком, но у неё его не оказалось с собой. Тогда мудрой черепахе пришлось пойти домой за ключиком по дорожке из больших кувшинок.

Начинает Тортилла в левом нижнем углу и может идти только на север и на восток, и ей стало интересно, сколько вариантов составить маршрут до правой верхней кувшинки у неё есть?

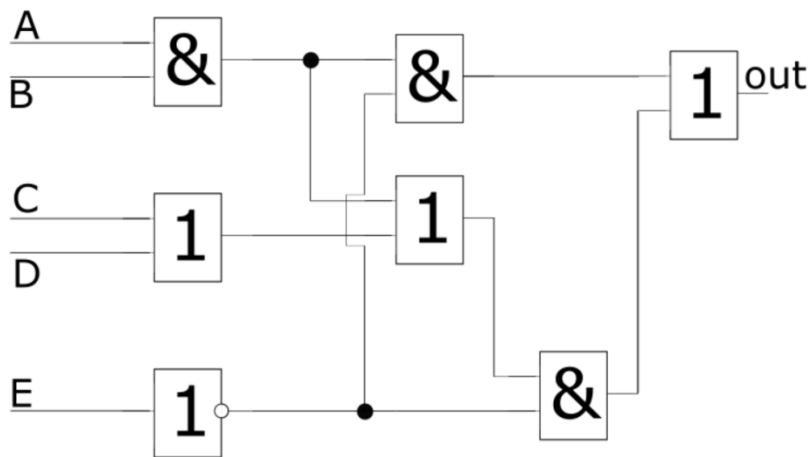


Ответ: 55

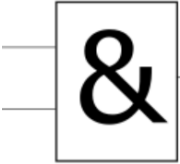
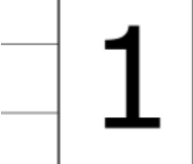
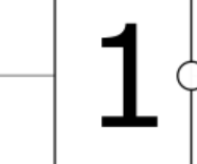
7. Основы логики. (2 балла)

[Логическая схема]

Дана логическая схема.



На схеме указаны обозначения следующих логических операций:

Название логической операции	Конъюнкция (И)	Дизъюнкция (ИЛИ)	Отрицание (НЕ)
Обозначение на схеме			

Примечание:

Для функции «не» известно, что если значение ее аргумента «ложь», то значение функции «истина» и наоборот.

Для функции «и» известно, что если все ее аргументы имеют значение «истина», то функция будет иметь значение «истина», при любых других сочетаниях значений аргументов функция принимает значение «ложь».

Для функции «или» известно, что если все ее аргументы имеют значение «ложь», то функция будет иметь значение «ложь», при любых других сочетаниях значений аргументов функция принимает значение «истина».

Сколько различных наборов аргументов (ABCDE), поданных на вход данной схеме, будут иметь на выходе истину?

Ответ: 13

8. Основы логики. (1 балл)

[Средневековая вечеринка]

Король Артур собрался праздновать окончание успешного похода за Граалем и пригласил на свою вечеринку 44 гостя. Некоторые из этих гостей лжецы, а некоторые – рыцари. Королю стало интересно, сколько среди гостей рыцарей. Первый гость сказал: «На пире меньше одного рыцаря», второй сказал: «На пире меньше двух рыцарей» и так далее, а 44-ый гость сказал: «На пире меньше 44 рыцарей». Сколько рыцарей было на пиру?

Ответ: 22

9. Кодирование информации. (3 балла)

[Взломщик Чак]

Алиса и Боб любят шпионское кино, и они решили придумать шифр, благодаря которому они смогут кодировать свои сообщения, чтобы никто не узнал, о чем они говорят.

Друг Алисы и Боба, Чак узнал, как работает шифр Алисы и Боба, и узнал, какой ключ используется.

Ключом является 4-значное число, записанное в десятичной системе счисления. Чак знает ключ, который будет в момент времени 00:00. Каждые 15 минут ключ меняется по следующему правилу:

Пусть X – ключ в момент времени суток 00:00, через 15 минут новый ключ

$$NEW_X = (X + 2764) \text{ MOD } 10000$$

Операция $A \text{ MOD } B$ вычисляет остаток от деления числа A на число B

Ещё через 15 минут за X будет взят действующий ключ и произведена та же операция с ключом.

Как происходит кодирование сообщений:

Исходный ключ, состоящий из 4 цифр – ABCD – делится на 2 пары цифр: AB и CD, образующих два новых числа.

Затем из каждого числа получается одно число: $A' = AB \text{ MOD } 26$, $B' = CD \text{ MOD } 26$

Полученные два числа соответствуют двум буквам латинского алфавита (нумерация букв в алфавите идет с нуля по возрастанию).

Далее к исходному сообщению применяется шифр, в ходе которого буква из алфавита под номером A' заменяется на букву из алфавита под номером B' , буква под номером $A'+1$ заменяется на букву под номером $B'+1$, и так далее. Если окажется, что номер очередной буквы больше, чем 25, то из него вычитается 26.

Пример:

В 00:00 ключ равен 1703, через 15 минут ключ поменяется на 4467.

$A' = 44 \text{ MOD } 26 = 18$, $B' = 67 \text{ MOD } 26 = 15$

Пусть исходное сообщение было ABACABA, тогда зашифрованное сообщение будет записано как XYXZXYX.

В 00:00 ключом является число 1925, а в 01:20 Чак перехватил сообщение Алисы с одним словом:

vszcc

Чаку стало интересно, что же Алиса написала Бобу. Помогите Чаку расшифровать данное сообщение. В ответе укажите только слово прописными латинскими буквами.

Ответ: hello

10. Алгоритмизация и программирование. (2 балла)

[Алгоритм с числом]

Вам дан алгоритм:

```
Алг
Нач
| цел: x = 31678969, s = 0, cnt = 0, t = 1, k = 0
|
| нц пока x >= 10
| | нц пока x != 0
| | | k = (x mod 10 + x div 10 mod 10) mod 10;
| | | s = s + k * t
| | | x = x div 100
| | | t = t * 10
| | кц
| | cnt = cnt + 1
| | x = s
| | s = 0
| | t = 1
| кц
| Вывод x, cnt
Кон
```

Операция $A \div B$ вычисляет результат целочисленного деления числа A на число B
Операция $A \bmod B$ вычисляет остаток от деления числа A на число B

В ответе укажите два числа через запятую без пробелов – результат работы данного алгоритма.

Ответ: 9,3

Отборочный этап. Второй тур (приведен один из вариантов заданий)

1. Теоретические основы информатики, единичный выбор. (1 балл).

[Операционные системы]

Какой из нижеперечисленных десертов никогда не использовался в качестве кодового имени версии Android?

1. KitKat
2. Snickers
3. Froyo
4. Oreo

Ответ: 2

2. Теоретические основы программирования, множественный выбор. (1 балл).

[Системы управления памятью]

Выберите все языки программирования, в которых используется сборщик мусора?

1. Python
2. Delphi
3. C#
4. C++
5. Pascal

Ответ: 1, 3

3. Алгоритмизация и программирование. Анализ кода. (2 балла).

[Заполнение массива]

Алексею очень нравится писать код на алгоритмическом языке, но у него нет под рукой исполнителя, чтобы его запустить. Он написал код, и ему стало интересно узнать, сколько чисел 9 будет записано в массиве, после окончания работы данного алгоритма:

```
a := 0
b := 1
c := 1
i := 0
k := 0
нц пока i < 100
  k := i
  нц пока i < k + b и i < 100
```

```

        A[i] := c
        i := i + 1
    КЦ
    c := c + 1
    b := b + a
    a := b - a
КЦ

```

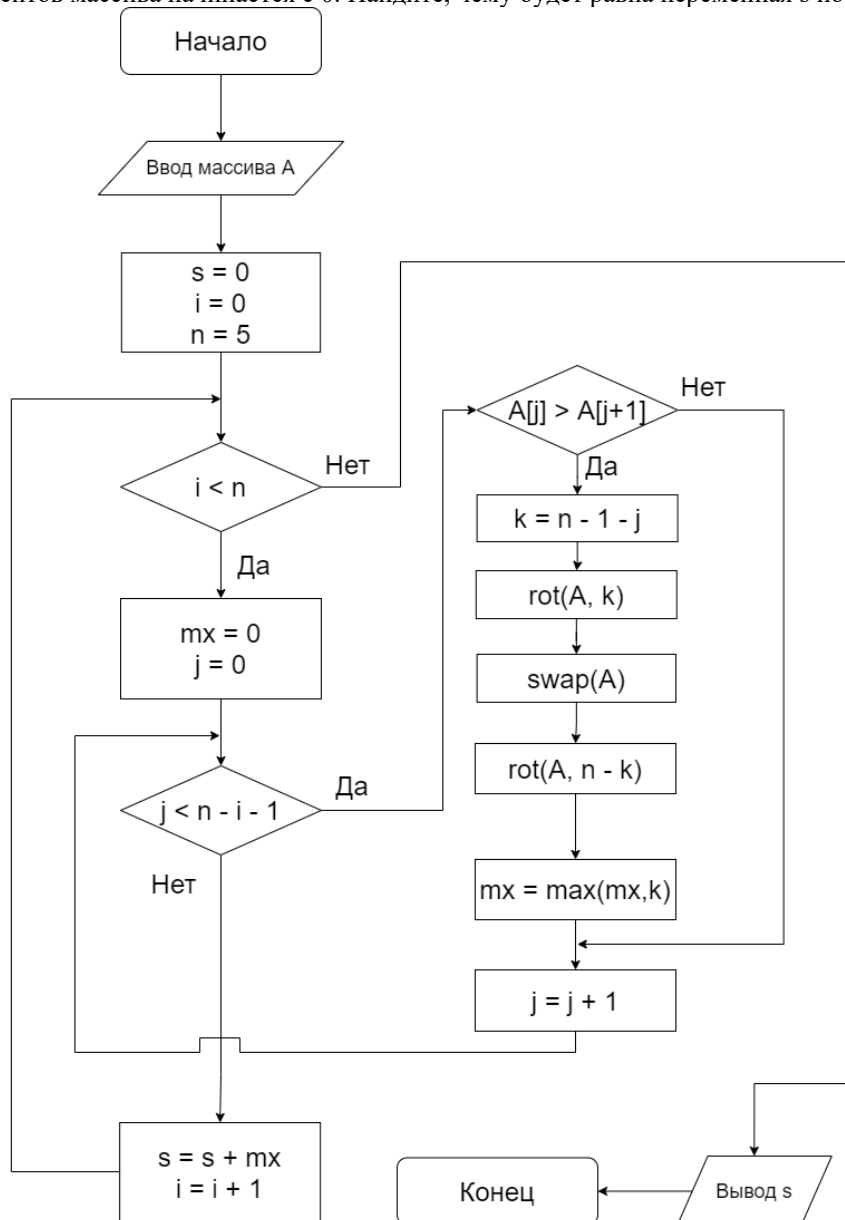
Изначально массив A имеет размер 100 элементов и заполнен нулями, нумерация элементов массива начинается с нуля. В ответе укажите одно число – искомое количество чисел 9 в массиве.

Ответ: 34

4. Алгоритмизация и программирование. Анализ блок-схем. (3 балла).

[От простого к сложному]

Для представленного ниже алгоритма известно, что на вход ему подали массив A = [85, 78, 53, 13, 32]. Нумерация элементов массива начинается с 0. Найдите, чему будет равна переменная s после окончания работы алгоритма:



Примечание:

Операция $\max(a, b)$ вычисляет максимальное из двух чисел

Операция $\text{rot}(A, k)$ делает циклический сдвиг массива вправо на k элементов

Операция $\text{swap}(A)$ меняет первый и последний элементы массива местами

В ответе укажите одно число – результат работы алгоритма.

Ответ: 12

5. Информационное моделирование. (1 балл).

[Оценка лабораторных работ]

В конце семестра добрый преподаватель по информатике, цифровой культуре и программированию задался вопросом, сможет ли он поставить всем своим студентам зачет, исходя из сданных ими лабораторных работ?

Известна информация о существующих баллах за лабораторные работы:

Предмет	Номер лабораторной	Количество баллов
Информатика	1	10
Цифровая культура	1	17
Программирование	1	X
Информатика	2	17
Цифровая культура	2	Y
Программирование	2	12
Информатика	3	20
Цифровая культура	3	10
Программирование	3	15
Информатика	4	13
Цифровая культура	4	15
Программирование	4	20
Информатика	5	Z
Цифровая культура	5	15
Программирование	5	10

А также информация о сданных лабораторных работах студентов:

Номер студента	Предмет, по которому студент сдавал лабораторную работу	Номера лабораторных работ, которые студент смог сдать по данному предмету
1	Информатика	1, 3, 4, 5
1	Программирование	1, 2, 3, 4, 5
1	Цифровая культура	2, 4, 5
2	Информатика	1, 2, 3, 5
2	Программирование	1, 3, 4, 5
2	Цифровая культура	1, 2, 4, 5
3	Информатика	1, 2, 3, 4, 5
3	Программирование	1, 2, 3
3	Цифровая культура	2, 3, 4, 5

Нужно найти такие минимальные баллы для каждой лабораторной работы с неизвестным числом баллов за неё, чтобы у всех трёх студентов по каждому предмету был зачёт. Зачёт выставляется, если у студента есть хотя бы 60 баллов после суммирования баллов за сданные ими лабораторные работы по предмету.

В ответе укажите через пробел три числа через пробел X Y Z, обозначающие количество баллов за лабораторные работы в указанном порядке.

Ответ: 33 30 17.

6. Информационное моделирование. (2 балла).

[Регулярные скороговорки]

Для задания регулярных выражений приняты следующие обозначения:

c	Любой неспециальный символ c соответствует самому себе. Специальными символами будем считать только символы [,] , { , } , * , + , - , ? – эти символы не могут по условию данной задачи встретиться в тексте.
[...]	Любой символ из ...; допустимы диапазоны типа: - a-z (последовательно идущие символы в алфавите), - A-Z (последовательно идущие символы в алфавите), 0-9 (последовательно идущие цифры). Диапазоны могут быть указаны друг за другом.
r*	Ноль или более вхождений символа r , может применяться и для диапазонов. Например, #* означает ноль или более вхождений символа # .
r+	Одно или более вхождений символа r , может применяться и для диапазонов, например [a-z]+ означает одно или более вхождений символов диапазона a-z в любом порядке.

r1r2	За символом или диапазоном r1 следует символ или диапазон r2 .
{a}	Число вхождений a предыдущего выражения. Например, выражение [a-z]{3} соответствует подстроке из трех латинских букв.
{a, b}	Число вхождений от a до b предыдущего выражения. Например, выражение [a-z]{3, 5} соответствует подстроке из не менее трех и не более пяти латинских букв.

Дан исходный текст.

These brothers bathe with those brothers, Those brothers bathe with these brothers. If these brothers did not bathe with those brothers, Would those brothers bathe with these brothers?

Его преобразовали таким образом, что каждое слово оказалось на отдельной строке, причем все знаки препинания были удалены, но различия в прописных и строчных буквах остались. Вам дано 5 регулярных выражений:

1. $[a-z]\{5\}$
2. $b[a-z]\{5\}r[a-z]$
3. $[c-z]^+$
4. $[0-9A-Z]^+[a-zA-z]$
5. $[a-z]^+h[a-z]\{3\}$

Выберите такое, которое позволит отобрать из преобразованного текста максимальное количество целых строк. Если таких регулярных выражений несколько, выберите любое из них. Для того, чтобы строка считалась отобранной, регулярное выражение должно соответствовать всей строке целиком, а не какой-либо её подстроке. Например, выражение $[a-c]\{2\}$ позволит отобрать строку ab, но не позволит отобрать строку dab.

Ответ: 5

7. Электронные таблицы. (2 балла).

[Опять геометрия]

Дан фрагмент электронной таблицы в режиме отображения формул:

B2													
=ЕСЛИ(И(B\$1*-1+\$A2*1-5 <= 0; B\$1*5+\$A2*2-10 <= 0; B\$1*-5+\$A2*12+10 >= 0);1;0)													
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	=СУММ(B2:V22)	=X1	=B1+1										
2	=X1	=ЕСЛИ(И											
3	=A2+1												
4													
5													
6													
7													

Значение ячейки B2 скопировали во все ячейки диапазона B2:V22. Значение ячейки C1 скопировали во все ячейки диапазона D1:V1. Значение ячейки A3 скопировали во все ячейки диапазона A4:A22. После перевода таблицы в режим отображения чисел значение ячейки A1 стало равно 42. Какое максимальное число могло быть записано в ячейку X1?

В ответе укажите целое(возможно отрицательное) число.

Ответ: -10

8. Алгоритмизация и программирование, информационное моделирование. (2 балла).

[Волшебные бобы]

Черепашка очень любит сладкие бобы. В один из дней она собрала из больших пустых банок квадрат 30 на 30 банок, и придумала алгоритм, по которому хочет положить в каждую банку бобы. Черепашка начинает с левой верхней банки и направлена в сторону правой верхней банки и действует по алгоритму:

1. Идёт прямо вдоль пустых банок и в каждую банку бросает по i бобов.
2. Как только дошла до края квадрата или перед ней стоит банка, в которой уже есть бобы, поворачивается направо на 90° .
3. Увеличивает i в 2 раза.
4. Возвращается к выполнению алгоритма с 1 пункта.

Черепашка заканчивает выполнения алгоритма заполнения банок тогда, когда во всех банках есть хотя бы один сладкий боб. В начале работы алгоритма $i = 1$.

Пример результата для поля из банок размером 4 на 4(числа обозначают количество бобов в банке):

1	1	1	1
8	16	16	2
8	64	32	2
4	4	4	2

Черепашке стало интересно, сколько на её поле после алгоритма получится банок, в которых ровно 1024 сладких боба.

В ответе укажите одно число – количество банок.

Ответ: 25

9. Моделирование по строке. (1 балл).

[Вихрь букв]

Даны три строки A, B и C:

Строка A = gjhkeyaое

Строка В = wectmrswnb

Строка С = abbe

Со строкой С десять раз совершаются следующие последовательности действий:

1. В конец строки С дописывается i -ый символ строки А.
2. В конец строки С дописывается i -ый символ строки В.
3. Полученная строка разворачивается.
4. К переменной i прибавляется единица.

В начале алгоритма $i = 0$, нумерация символов строках начинается с нуля.

Чему будет равен символ с индексом 20 в строке С после выполнения всех действий?

В ответе укажите прописную букву латинского алфавита – искомый символ.

Ответ: у

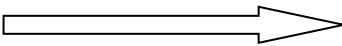
10. Информационное моделирование. (3 балла).

[Любители симуляторов]

Петя и Вася очень любят различные симуляции. Как-то раз им пришла в голову идея задать правила для симуляции изменения поля клеток размером 3 на 4 ячейки. В каждую ячейку они поместили по одному целому числу, и задали правила для симуляции: за одну эпоху каждая ячейка с положительным числом увеличивает все соседние с ней по стороне ячейки на это самое положительное число. Симуляция завершается тогда, когда все числа в ячейках поля становятся неотрицательными.

Пример одной эпохи для поля $3 * 4$:

1	-2	3	-4
5	-6	7	-8
9	-10	11	-12



6	2	10	-1
15	6	21	-1
14	10	18	-1

Помогите Пете и Васе для заданного поля определить, сколько эпох должно пройти, чтобы симуляция завершилась?

-805	90	-986	-892
-575	-731	-748	76
-712	-574	-174	-901

В ответе укажите одно число – искомый ответ на вопрос Пети и Васи.

Ответ: 14