

I. Задания заключительного этапа олимпиады 2018-19 года

Заключительный этап 11 класса (приведен один из вариантов заданий)

1. Кодирование информации. Системы счисления (2 балла)

[Единицы]

Сколько единиц в двоичной записи числа R, если известно, что

$$R = \frac{0, B6DB6DB6DB6D_{16}}{0, A_{16}}$$

Учитываются единицы как в целой, так и в дробной части числа R. В ответе укажите целое число.

Ответ: 16

Решение:

Поскольку вопрос задания связан с двоичной системой счисления, переведем числитель и знаменатель дроби в эту систему счисления. Обратим внимание, что в числителе в дробной части 4 раза последовательно повторяется фрагмент B6D₁₆. Если перевести его в двоичную систему счисления (напомним, что для этого необходимо каждой шестнадцатеричной цифре поставить в соответствие четыре двоичных разряда), получим, что B6D₁₆=101101101101₂. Обратим внимание, что это записанные подряд четыре фрагмента 101₂. Тогда вся дробная часть числа в числителе в двоичной записи будет представлять собой 16 записанных подряд фрагментов 101₂. Переведем в двоичную систему счисления знаменатель, получим 0, A₁₆=0,101₂. Таким образом, легко заметить, что дробная часть числа в числителе - это записанные последовательно 16 раз все цифры из дробной части знаменателя.

Рассмотрим, как с помощью операции умножения можно получить из числа 0,101₂ число 0,101101₂.

$$0,101101_2 = 0,101_2 + 0,000101_2 = 0,101_2 + 0,101_2 * 0,001_2 = 0,101_2 * 1,001_2$$

Рассуждая аналогично можно получить, что 0,101101101₂=0,101₂*1,001001₂ и т.д.

Таким образом, число R это число, целая часть которого равна 1, а дробная часть состоит из последовательности из (n-1) двоичных фрагментов 001₂, где n - количество фрагментов 101₂ в дробной части числа в числителе. Следовательно, число R содержит 16 единиц в своей записи.

2. Кодирование информации. Объем информации (3 балла)

[Буферизация]

Петя и Вася разрабатывают приложения для сохранения в памяти текстовых данных. Приложения работают по одинаковому алгоритму. Они получают на вход потоки символов, в которых каждый символ с равной вероятностью выбирается из алфавита мощностью 151 символ. Поступающие символы собираются в буфере определенного размера. Как только буфер полностью заполняется, приложение кодирует содержимое буфера, однозначно сопоставляя получившейся в буфере комбинации символов одно число и записывая в память код этого числа, используя для записи минимальное, одинаковое для всех возможных таких чисел количество бит.

Петя решил использовать буфер размером N символов, а Вася – буфер размером M символов. Каждый из них сохранил входной поток размером в 4488 символов, и обнаружилось, что для этого Пете потребовалось ровно на 232 бита больше, чем Васе.

Известно, что ни Петя, ни Вася не использовали буферы, размером более чем 50 символов, а также, что в результате получения последних символов потока и у Пети, и у Васи буфер оказывался заполнен полностью.

Определите N и M, при которых это возможно. В ответе укажите через пробел два целых числа – сначала N, затем M.

Ответ: 17 33

Решение:

Построим модель. В буфере размером в N символов при условии использования алфавита из 151 символа может получиться одна из 151^N комбинаций символов. Тогда для записи в память одного из 151^N чисел так, чтобы для любого числа требовалось минимальное одинаковое число бит, необходимо использовать log₂(151^N) бит. Обратим внимание, что поскольку нам необходимо сохранять целое число бит, вычисленное значение логарифма должно быть округлено до ближайшего большего целого числа. Тогда для сохранения всего потока приложением Пети потребуется P=4488/N* log₂(151^N) бит. Аналогично для сохранения всего потока приложением Васи потребуется V=4488/M* log₂(151^M) бит.

Соответственно, нам нужно подобрать такие N и M, чтобы разность между P и V была ровно 232 бита. Также необходимо учесть ограничения задачи: N, M <= 50, а также N и M являются делителями числа 4488 (поскольку по условию последний символ потока в обоих случаях приводил к полному заполнению буфера).

Первый вариант – решить уравнение. Обозначим log₂(151^N) с округлением в большую сторону до ближайшего целого за Im. Обозначим log₂(151^M) с округлением в большую сторону до ближайшего целого за In. Тогда условие задачи может быть описано уравнением 4488*In/N - 4488*Im/M=232

Сократим левую и правую часть на 8 и разложим на множители:

$$3*11*17*(M*In-N*Im)/(N*M)=29$$

Все числа в уравнении – целые. 29 – простое число. Предположим, что 3*11*17 = N*M. Поскольку N, M <=50, получается вариант, что N=33 и N=17 или M=17 и N=33. Проверим оба варианта и получим, что при N=17 и M=33

$$33*log_2(151^{17}) - 17*log_2(151^{33}) = 29. \text{ Следовательно, мы правильно определили, что } N=17 \text{ и } M=33.$$

Также, подбор N и M, удовлетворяющих построенной модели, может быть выполнен с использованием любого языка программирования, но нужно учесть необходимость работы с большими числами. Например, на Python программа для подбора значений может выглядеть так:

```
import math
for N in range(1, 50):
    for M in range(1, 50):
```

```

if 4488%(M*N)==0:
    P=4488/N*math.ceil(math.log2(151**N))
    V=4488/M*math.ceil(math.log2(151**M))
    if P-V==232:
        print(N,M)

```

В результате получается единственный ответ N=17, M=33.

3. Основы логики (2 балла)

[Логический преобразователь]

Петя написал логический преобразователь. На вход он получает целое положительное восьмиразрядное двоичное число вида: $X_7X_6X_5X_4X_3X_2X_1X_0$, где X_7 - старший разряд числа, а X_0 - младший разряд числа.

В преобразователе реализованы две логические функции:

$$F_1 = ((X_0 \text{ xor } X_1) \rightarrow X_2) \vee ((X_3 \text{ xor } X_4) \rightarrow \overline{X_5}) \vee ((X_6 \text{ xor } X_7) \rightarrow X_2)$$

$$F_2 = ((X_0 \text{ xor } X_5) \rightarrow \overline{X_7}) \vee ((X_1 \text{ xor } X_4) \rightarrow X_6) \vee ((X_3 \text{ xor } X_4) \rightarrow X_2)$$

Значения разрядов X_i , равные единице считаются за истинные значения одноименных логических переменных, а равные нулю - за ложные.

На выходе преобразователя получаются значения двух функций.

Известно, что если подать на вход некоторое число N, то на выходе получатся значения обеих функций равные "ложь".

Найдите это число N и запишите его в ответ в десятичной системе счисления.

Ответ: 170

Решение:

Поскольку обе функции принимают ложные значения, а каждая из них представляет собой дизъюнкцию трех выражений в скобках, можно сделать вывод, что все выражения в скобках должны принимать ложные значения. Поскольку импликация может принимать ложное значение только, если ее правая часть ложна, мы можем однозначно определить значения четырех переменных: $X_2=0$, $X_5=1$, $X_6=0$, $X_7=1$. Левые части импликаций должны быть истинны. Тогда, поскольку $X_5=1$, а в первой скобке второй функции $(X_0 \text{ xor } X_5)$ должно быть истинным, $X_0=0$. Аналогично, исходя из того, что в первой скобке первой функции $(X_0 \text{ xor } X_1)$ должно быть истинным, $X_1=1$. Теперь можно определить значение X_4 . Исходя из второй скобки во второй функции $(X_1 \text{ xor } X_4)$ должно быть истинным. Следовательно, $X_4=0$. Осталось определить значение X_3 . Исходя из второй скобки первой функции $(X_3 \text{ xor } X_4)$ должно быть истинным. Следовательно, $X_3=1$.

Значит искомое число $N=10101010_2 = 170_{10}$

4. Кодирование информации. Алгоритмы обработки кодированной информации (1 балл)

[Не-Лишрел]

Дан алгоритм построения палиндрома. На вход подается натуральное число A. Если оно не является палиндромом, то переворачивается (цифры числа записываются в обратном порядке) и результат складывается с исходным числом. Если полученная сумма не является палиндромом, то ее цифры вновь записываются в обратном порядке и результат складывается с этой суммой. Такой процесс продолжается до тех пор, пока после очередного сложения не будет получен палиндром.

Большинство чисел за конечное число итераций такого алгоритма приводят к палиндрому. Для некоторых чисел это свойство еще не доказано, и такие числа называются числами Лишрел.

Найдите наименьшее число A, для которого ровно за три итерации в результате работы представленного алгоритма получено число $R=11011$, являющееся палиндромом.

Пример:

Для заданного числа $R = 2552$ искомое число $A=184$.

Проверяем:

$$184 + 481 = 665$$

$$665 + 566 = 1231$$

$$1231 + 1321 = 2552 \text{ - палиндром!}$$

В ответе укажите натуральное число.

Ответ: 158

Решение:

Приведем пример программы на языке Python для поиска требуемого числа:

```

def revert(N) :
    result=0
    while N>0 :
        result=result*10+N%10
        N=N//10
    return result

Z=11011
for i in range(10, Z) :
    if revert(i) != i :
        result1=i+revert(i)
        if revert(result1) != result1 :

```

```

result2=result1+revert(result1)
if revert(result1) != result2:
    result3=revert(result2)+result2
    if result3==Z:
        print(i)
        exit(0)

```

Можно решить задачу и без программирования. 11011 можно получить при сложении следующих пар чисел: 5006+6005; 4007+7004; 3008+8003; 2009+9002; 10010+01001;

Рассмотрение всех этих вариантов показывает, что только число 10010 может быть получено сложением следующих чисел, которые удовлетворяют условию задачи:

1009+9001.

В свою очередь $1009=158+851$

5. Алгоритмизация и программирование. Формальные исполнители (1 балл)

[А и В]

Исходная строка содержит 24 символа и выглядит следующим образом:

'AABVVAABVVAABVVAABVVAABVVAABV'

Есть цикл ее обработки. На каждом шаге цикла осуществляются следующие операции:

1. Копия строки, полученной на предыдущем шаге, дописывается ей в конец.
2. Отсчитывается N-ый символ 'A', считая с конца строки, и удаляются все символы, начиная с него и до конца строки.

Известно, что после завершения 15-го шага цикла в строке оказалось ровно 163847 символов 'A'. Определите, при каком N это возможно. В ответе укажите целое положительное число.

Ответ: 7

Решение:

Проанализируем, как меняется количество букв А на после выполнения очередного шага алгоритма. Обратим внимание, что на каждом шаге алгоритма количество букв А сначала удваивается, а затем уменьшается на N. Обозначим за X_i – количество букв А на i-том шаге.

$$X_0=12$$

$$X_1=2*X_0-N$$

$$X_2=2*X_1-N=2*(X_0-N)-N=4*X_0-3*N$$

$$X_3=2*X_2-N=2*(4*X_0-3*N)-N=8*X_0-7*N$$

$$X_4=2*X_3-N=2*(8*X_0-7*N)-N=16*X_0-15*N$$

$$\text{Тогда } X_n=2^n*X_0-(2^n-1)*N$$

Следовательно, по условию задания можно составить уравнение:

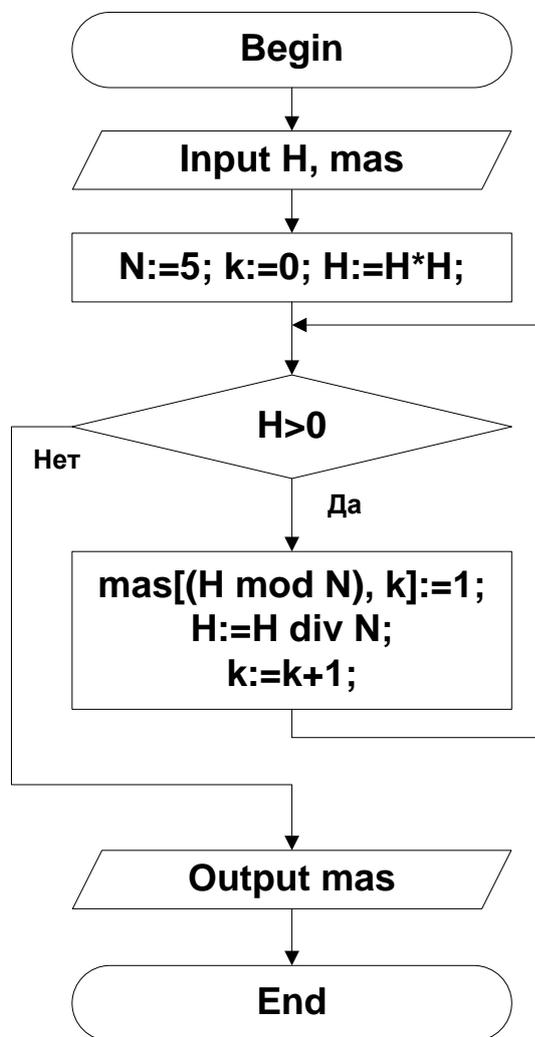
$$2^{15}*12-(2^{15}-1)*N=163847$$

Решив уравнение получаем, что $N=7$.

6. Алгоритмизация и программирование. Анализ алгоритма, заданного в виде блок-схемы (3 балла)

[Галочка]

Дана блок-схема алгоритма, обрабатывающего двумерный массив. На вход подается массив **mas**, размером 5×7 элементов, инициализированный нулями и целое положительное число **N**. Нумерация элементов массива начинается с [0,0].



Найдите такое минимальное число H , чтобы на выходе получился массив:

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

При обращении к элементам массива первый индекс обозначает номер строки, а второй – номер столбца.

Операция $A \bmod B$ вычисляет остаток от целочисленного деления A на B . Операция $A \operatorname{div} B$ вычисляет частное от целочисленного деления A на B . В ответе укажите целое число.

Ответ: 156

Решение:

Проанализируем алгоритм. Поступившее на вход число H возводится в квадрат, а затем обрабатывается в цикле. Обратим внимание, что на каждом шаге вычисляется остаток от деления числа H на 5, а затем H нацело делится на 5. Цикл продолжается пока H остается положительным числом. Таким образом, мы последовательно получаем разряды пятеричной записи числа H . Переменная k , являющаяся номером столбца, последовательно увеличивается на единицу, с начального значения 0. Следовательно, на каждом шаге цикла устанавливается значение 1 для элемента массива, номер строки которого означает очередную цифру пятеричной записи числа H , а номер столбца – порядковый номер этой цифры в записи числа, считая справа налево. Тогда выводимый в результате массив позволяет сделать вывод, что $H^2 = 1234321_5$. Легко заметить, что такое число может быть получено путем возведения в квадрат 1111_5 . Осталось перевести число в десятичную систему счисления и получить 156_{10} .

7. Телекоммуникационные технологии (2 балла)

[Подсети]

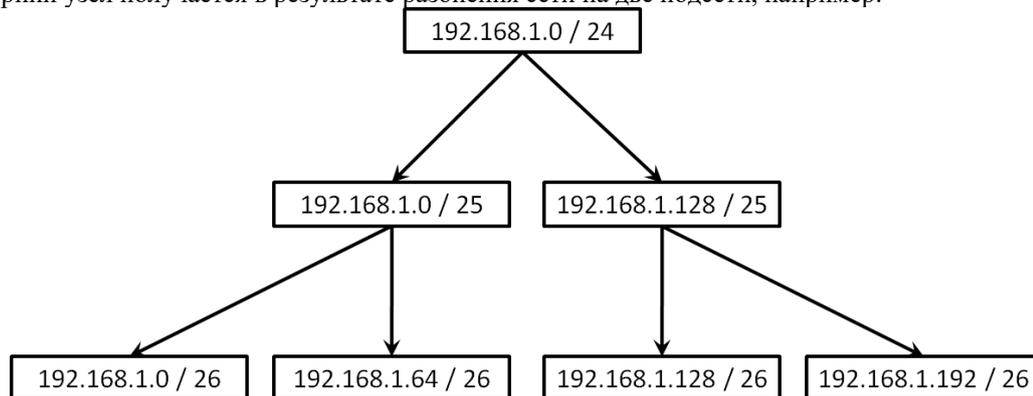
Маска сети для IPv4 адресации – это 4-х байтное число, которое делит IP адрес на адрес сети (первая часть) и адрес узла (вторая часть). У всех адресов одной IP-сети совпадают первые части и отличаются вторые. Для части IP-адреса, соответствующей адресу сети, в маске сети содержатся двоичные единицы, а для части IP-адреса, соответствующей адресу узла, в маске сети содержатся двоичные нули. Для записи масок сетей часто используется нотация, когда после IP-адреса через «/» указывается число бит, отводимых в маске под адрес сети. Например, для адреса 11.12.0.8 и маски 255.0.0.0 запись будет иметь следующий вид 11.12.0.8/8. Такое число, которое указывается после «/», называется префиксом сети.

Помощнику сетевого администратора корпоративной сети поставлена задача: разделить сеть нового филиала компании на подсети в соответствии с определенным подходом. На филиал выделяется общее адресное пространство. Необходимо

обеспечить каждому отделу филиала требуемое ему количество IP-адресов для клиентских устройств в отделе, возможно с некоторым запасом.

Подход к разделению сетей выглядит следующим образом. Подсеть разбивается на более мелкие подсети увеличением префикса подсети. Поскольку префикс задаёт количество бит в части IP-адреса, соответствующей адресу сети, такая операция всегда делит общее количество адресов подсети, на число, кратное степени двойки.

Исходную сеть и её подсети можно представить в виде графа, вершиной которого будет исходная сеть, а каждый дочерний узел получается в результате разбиения сети на две подсети, например:



и т.д.

Таким образом, процесс разбиения исходной сети на нужное количество подсетей сводится к дихотомическому делению исходной сети до тех пор, пока для каждого отдела не будет найдена подсеть **минимально необходимого** размера. Следует отметить, что в каждой подсети есть два адреса, которые не могут быть назначены клиентским устройствам отдела, но входят в выделяемый для подсети диапазон адресов – это адрес сети и широковещательный адрес. При этом отделу с **большим** количеством IP-адресов будет выделяться подсеть с **меньшим** адресом сети, а при соблюдении этого требования каждому отделу будет выделяться подсеть с минимально возможным адресом сети.

Например, пусть требуется разделить исходное адресное пространство так, чтобы выделить подсети трем отделам с указанным количеством компьютеров в каждом отделе:

| Адресное пространство (блок адресов) | Количество клиентских устройств в отделах | | |
|--------------------------------------|---|---|----|
| | А | В | С |
| 192.168.1.0 /24 | 28 | 7 | 46 |

Тогда будут выделены адреса сетей: С – 192.168.1.0/26, А – 192.168.1.64/27, В – 192.168.1.96/28.

Широковещательный адрес – последний адрес в диапазоне подсети: для С - 192.168.1.63/26, А - 192.168.1.95/27, В - 192.168.1.111/28.

Задача:

Применить указанный подход и указать в ответе широковещательный адрес подсети А, если известно:

| Адресное пространство (блок адресов) | Количество клиентских устройств в отделах | | |
|--------------------------------------|---|----|----|
| | А | В | С |
| 36.121.96.0 /24 | 8 | 51 | 17 |

Пример записи ответа: 192.168.1.63/26

Ответ: 36.121.96.111/28

Решение:

Обратим внимание, что подсеть с префиксом /24 позволяет выдать адреса $2^{(32-24)}-2=254$ клиентским устройствам, а подсеть с префиксом /25 – $2^{(32-25)}-2=126$ клиентским устройствам. Подсеть с префиксом /26 позволит выдать адреса $2^{(32-26)}-2=62$ клиентским устройствам, то есть это минимальная подсеть для отдела В. Поскольку наиболее крупному отделу нужно отдать подсеть с меньшим адресом сети – отдел В займет адреса в диапазоне 36.121.96.0/26 - 36.121.96.63/26. Диапазон, начинающийся с 36.121.96.64/24, будет свободен.

Подсеть с префиксом /27 позволит выдать адреса $2^{(32-27)}-2=30$ клиентским устройствам, то есть это минимальная подсеть для отдела С. Тогда ему будет выделен диапазон адресов 36.121.96.64/27 - 36.121.96.95/27.

Подсеть с префиксом /28 позволит выдать адреса $2^{(32-28)}-2=14$ клиентским устройствам, то есть это минимальная подсеть для отдела А (обратим внимание, что если мы проведем еще одно деление, то подсеть с префиксом /29 позволит использовать для клиентских устройств только 6 адресов). Тогда для этой подсети будет выделен диапазон 36.121.96.96/28 - 36.121.96.111/28. Последний адрес этого диапазона будет широковещательным адресом и должен быть записан в ответе.

8. Технологии обработки информации в электронных таблицах (2 балла)

[Матрица]

Дан фрагмент электронной таблицы в режиме отображения формул:

| | A | B | C | D |
|---|-------|---|-------|---|
| 1 | | 0 | =B1+1 | |
| 2 | 0 | =ОСТАТ(B\$1+\$A2*(МАКС(\$A\$2:\$A\$80)+1);\$A\$1) | | |
| 3 | =A2+1 | | | |
| 4 | | | | |

Ячейку C1 скопировали во все ячейки диапазона D1:CB1. Ячейку A3 скопировали во все ячейки диапазона A4:A80. Ячейку B2 скопировали во все ячейки диапазона B2:CB80. Сколько существует целых положительных чисел, таких что если поместить такое число в ячейку A1, то во всех ячейках диагонали B2, C3, D4, E5, ..., CA79, CB80 окажутся только нулевые значения? В ответе укажите целое число.

Ответ: 10

Решение:

Обратим внимание, что в первой строке, начиная с ячейки B1 и до ячейки CB1, диапазон будет последовательно заполнен целыми числами от 0 до 78. Аналогично в столбце A диапазон A2:A80 будет последовательно заполнен целыми числами от 0 до 78, то есть максимальное из них - 78. Тогда часть формулы $B\$1 + \$A2 * (\text{МАКС}(\$A\$2:\$A\$80) + 1)$ после копирования будет давать последовательность целых чисел от 0 до $79^2 - 1$ для ячеек, отсчитываемых построчно слева направо с переходом на следующую строку после последней ячейки в столбце CB. Тогда полная формула, записанная в ячейке B2 после копирования будет заполнять ячейки диапазона B2:CB80 слева направо в строке, построчно сверху вниз остатками от деления чисел от 0 до $79^2 - 1$ на число в ячейке A1. Легко заметить, что эти остатки будут образовывать цикл 0, 1, ..., [число, записанное в ячейке A1], 0, 1, и т.д. Обратим внимание, что в заполняемой матрице длина строки составляет 79 ячеек. Тогда при последовательном заполнении слева направо в строке, построчно сверху вниз, элементом, находящимся на главной диагонали будет каждый 80-ый элемент. Следовательно, чтобы он был равен нулю, должен быть равен нулю остаток от деления 80 на число в ячейке A1. Тогда задача сводится к определению делителей числа 80. Таких делителей ровно 10: 1, 2, 4, 5, 8, 10, 16, 20, 40, 80.

9. Технологии сортировки и фильтрации данных (1 балл)

[Конкуренты]

Есть таблица с заказами на продажу товаров. Каждая строка таблицы - один заказ на продажу товара. В каждом заказе один товар.

| IDТовара | Цена 1, тыс. руб. | Цена 2, тыс. руб. |
|----------|-------------------|-------------------|
| 1 | 5 | 9 |
| 2 | 5 | 6 |
| 3 | 10 | 8 |
| 4 | 6 | 10 |
| 5 | 8 | 7 |
| 6 | 4 | 6 |
| 7 | 7 | 8 |
| 8 | 6 | 5 |
| 9 | 6 | 4 |
| 10 | 5 | 8 |
| 11 | 9 | 6 |
| 12 | 6 | |

Для каждого товара указаны две цены, соответственно в столбцах "Цена 1" и "Цена 2". Петя и Вася выбирают заказы из таблицы и выполняют их в соответствии со следующими правилами:

1. Петя продает товары по цене, указанной в столбце "Цена 1", а Вася - по цене, указанной в столбце "Цена 2".
2. Петя и Вася по очереди выбирают из таблицы по два заказа так, чтобы получить при их выполнении максимальную сумму. Если таких пар несколько, может быть выбрана любая из них.
3. Петя начинает первым.
4. Выполненные заказы удаляются из таблицы.

Определите значение в закрашенной ячейке таблицы, если известно, что после выполнения всех заказов Вася заработал на одну тысячу рублей больше, чем Петя. В ответе укажите целое число.

Ответ: 8

Решение:

Пусть в указанной ячейке число 10. Тогда порядок продаж будет следующим (выделены ячейки с ценами, по которым осуществляются продажи в порядке выполнения этих продаж):

| IDТовара | Цена 1, тыс. руб. | Цена 2, тыс. руб. |
|----------|-------------------|-------------------|
| 3 | 10 | 8 |
| 11 | 9 | 6 |

| | | |
|----|---|----|
| 4 | 6 | 10 |
| 12 | 6 | 10 |
| 5 | 8 | 7 |
| 7 | 7 | 8 |
| 1 | 5 | 9 |
| 10 | 5 | 8 |
| 8 | 6 | 5 |
| 9 | 6 | 4 |
| 2 | 5 | 6 |
| 6 | 4 | 6 |

Легко заметить, что в этом случае Вася заработает на 3 тысячи больше, чем Петя.

Теперь рассмотрим вариант, что в указанной ячейке число 9. В этом случае возможны два варианта

| IDТовара | Цена 1, тыс. руб. | Цена 2, тыс. руб. |
|----------|-------------------|-------------------|
| 3 | 10 | 8 |
| 11 | 9 | 6 |
| 4 | 6 | 10 |
| 1 | 5 | 9 |
| 5 | 8 | 7 |
| 7 | 7 | 8 |
| 12 | 6 | 9 |
| 10 | 5 | 8 |
| 8 | 6 | 5 |
| 9 | 6 | 4 |
| 2 | 5 | 6 |
| 6 | 4 | 6 |

| IDТовара | Цена 1, тыс. руб. | Цена 2, тыс. руб. |
|----------|-------------------|-------------------|
| 3 | 10 | 8 |
| 11 | 9 | 6 |
| 4 | 6 | 10 |
| 12 | 6 | 9 |
| 5 | 8 | 7 |
| 7 | 7 | 8 |
| 1 | 5 | 9 |
| 10 | 5 | 8 |
| 8 | 6 | 5 |
| 9 | 6 | 4 |
| 2 | 5 | 6 |
| 6 | 4 | 6 |

Как видно, в обоих случаях Вася заработает на 2 тысячи рублей больше, чем Петя.

Рассмотрим вариант, когда в указанной ячейке число 8.

| IDТовара | Цена 1, тыс. руб. | Цена 2, тыс. руб. |
|----------|-------------------|-------------------|
| 3 | 10 | 8 |
| 11 | 9 | 6 |
| 4 | 6 | 10 |
| 1 | 5 | 9 |
| 5 | 8 | 7 |
| 7 | 7 | 8 |
| 10 | 5 | 8 |
| 12 | 6 | 8 |
| 8 | 6 | 5 |
| 9 | 6 | 4 |
| 2 | 5 | 6 |
| 6 | 4 | 6 |

Как видно из таблицы, в этом случае Вася заработает ровно на 1 тысячу больше, чем Петя и ответ 8 нам подходит.

Тем не менее, рассмотрим варианты с меньшими значениями в закрашенной ячейке. Обратим внимание, что в этом случае первые четыре продажи Пети и первые две продажи Васи не могут изменить свой порядок.

| IDТовара | Цена 1, тыс. руб. | Цена 2, тыс. руб. |
|----------|-------------------|-------------------|
| 3 | 10 | 8 |
| 11 | 9 | 6 |
| 4 | 6 | 10 |
| 1 | 5 | 9 |
| 5 | 8 | 7 |
| 7 | 7 | 8 |
| 10 | 5 | 8 |

| | | |
|----|---|---|
| 12 | 6 | 7 |
| 8 | 6 | 5 |
| 9 | 6 | 4 |
| 2 | 5 | 6 |
| 6 | 4 | 6 |

Как видно из таблицы, Петя и Вася заработают одинаково в этом случае.

Легко показать, что при меньших значениях в закрашенной ячейке у Васи не остается возможностей заработать больше, чем Пете. Следовательно, ответ единственный и это 8.

10. Технологии программирования (2 балла)

[Свойства объектов]

| | |
|------------------------|--------------|
| Имя входного файла | input.txt |
| Имя выходного файла | output.txt |
| Ограничение по времени | 2 секунды |
| Ограничение по памяти | 256 мегабайт |

В хранилище Васи находится n объектов, пронумерованных от 1 до n , у каждого из которых есть некоторое количество свойств (возможно, ни одного). Каждое свойство представлено в виде натурального числа от 1 до 10^9 .

Проанализировав устройство своего хранилища, Вася решил, что оно должно поддерживать две операции:

- Удаление устаревшего свойства s . При удалении свойства, оно удаляется у всех объектов, которым принадлежит. Если указанного свойства не существует, ничего делать не нужно.
- Найти количество оставшихся свойств у объекта с номером r .

Васе очень нужно реализовать эту функциональность, и он обратился к вам за помощью. Помогите ему - напишите программу, которая будет поддерживать обе операции, нужные Васе.

Формат входных данных

В первой строке входного файла содержится число n - количество объектов в хранилище Васи ($1 \leq n \leq 10^5$). В i -й из следующих n строк содержится описание свойств объекта с номером i : сначала дано число k_i - количество свойств у i -го объекта, а затем через пробел даны k_i чисел $p_{i,j}$ - свойства i -го объекта ($0 \leq k_i \leq 100$, $1 \leq p_{i,j} \leq 10^9$).

Все объекты пронумерованы от 1 до n в порядке, представленном во входных данных. Гарантируется, что общее количество свойств у всех объектов не превосходит 10^5 . Также гарантируется, что для каждого i все $p_{i,j}$ различны.

В $n + 2$ строке содержится число q - количество запросов к хранилищу Васи ($1 \leq q \leq 10^5$).

В j -й из следующих q строк содержится информация об j -м запросе:

- c , если из хранилища требуется удалить устаревшее свойство c ($1 \leq c \leq 10^9$);
- ? r , если требуется найти количество оставшихся свойств у объекта с номером r .

Формат выходных данных

Для всех запросов на нахождение количества оставшихся свойств у объекта, в отдельных строках, в порядке их поступления для каждого запроса выведите это количество.

Пример

| input.txt | output.txt |
|-----------|------------|
| 2 | 2 |
| 3 1 2 4 | 3 |
| 3 2 3 5 | 1 |
| 12 | 2 |
| - 1 | 1 |
| ? 1 | 1 |
| ? 2 | 1 |
| - 2 | 1 |
| ? 1 | |
| ? 2 | |
| - 5 | |
| ? 1 | |
| ? 2 | |
| - 6 | |
| ? 1 | |
| ? 2 | |

Замечание

Свойство 1 есть только у первого объекта, поэтому после его удаления у первого объекта остается 2 свойства, а у второго все еще 3.

Свойство 2 есть у обоих объектов, поэтому оно удаляется у обоих объектов, у первого объекта остается 1 свойство, а у второго - 2.

Свойство 5 есть только у второго объекта, поэтому после его удаления у обоих объектов остается 1 свойство.

Свойства 6 нет ни у одного объекта, поэтому его удаление не меняет количество свойств у объектов.

Решение:

Для эффективного решения этой задачи требуется правильно выбрать способ хранения данных. Если у каждого объекта хранить множество его свойств, то:

- Операция удаления свойства с из хранилища будет выполняться за $O(n)$ (надо пройти по всем объектам и удалить у них свойство с, если оно есть).
- Операция получения количества оставшихся свойств у объекта будет работать за $O(1)$.

Такое решение в худшем случае работает за $(n \cdot q)$, что не укладывается по времени.

Оптимизируем наше решение: у каждого свойства с будем хранить список объектов obj_c , которым оно принадлежит, а для каждого объекта г будем поддерживать оставшееся количество свойств $remaining_g$:

Операция удаления свойства с из хранилища будет работать за $O(m)$, где m — количество объектов, у которых есть свойство с: надо взять все объекты, у которых есть свойство с (эта информация у нас сохранена в obj_c) и у всех них уменьшить количество оставшихся свойств на 1. После этого из obj_c надо удалить все элементы.

Операция получения оставшегося количества свойств у объекта г будет работать за $O(1)$: эта информация хранится в $remaining_g$.

Несложно заметить, что суммарная асимптотика нового алгоритма не превышает суммарного количества свойств у всех объектов, что, по условию, не больше 10^5 . Такой алгоритм уже спокойно уложится в ограничение по времени.

11. Технологии программирования (4 балла)

[Оптимальное подмножество строк]

| | |
|------------------------|--------------|
| Имя входного файла | input.txt |
| Имя выходного файла | output.txt |
| Ограничение по времени | 2 секунды |
| Ограничение по памяти | 256 мегабайт |

Сегодня на уроке Петя узнал про беспрефиксные коды. Множество строк называется беспрефиксным кодом, если

- Все строки в множестве различны
- Не существует такой пары различных строк, что одна строка является префиксом другой

Строка s называется префиксом строки t , если длина строки s не больше длины t , а также для любого i , i -й символ строки s совпадает с i -м символом строки t . Например, ab является префиксом ab и abc , но не является префиксом a и ac .

Петя же решил придумать что-то новое и ввел новое понятие — k -беспрефиксный код. Таким кодом он назвал множество строк, такое, что:

- Все строки в множестве различны
- У любых двух различных строк наибольший общий префикс имеет длину не больше k

Наибольшим общим префиксом двух строк s и t называется наибольшая по длине строка, являющаяся префиксом обеих строк.

Теперь по данному множеству строк s_1, s_2, \dots, s_n и числу k Петя хочет найти в этом множестве k -беспрефиксный код, состоящий из максимально возможного количества строк. Помогите ему — найдите этот код.

Входные данные

В первой строке входного файла содержится два числа n и k — количество строк в множестве и максимальная длина общего префикса соответственно ($1 \leq n \leq 10^5, 1 \leq k \leq 100$).

В i -й из следующих n строк содержится строка из строчных латинских букв s_i — i -е слово из множества ($1 \leq |s_i| \leq 100$). Гарантируется, что суммарная длина всех строк в множестве не превосходит 10^6 .

Выходные данные

В первой строке выходного файла выведите число m - максимально возможное количество строк в k -беспрефиксном коде. В i -й из следующих m строк выведите i -й элемент этого кода. Элементы кода можно выводить в любом порядке.

Если существует несколько ответов с максимальным m , разрешается вывести любой.

Примеры

| | |
|---|-------------------------|
| input.txt | output.txt |
| 5 2 aabc acbd aac acba aab | 3 aab aac acba |
| 4 2 aa abc | 3 aa abb |

| | |
|-----|-----|
| aa | abc |
| abb | |

Примечание

В первом примере у строк 1 и 5, а также у строк 2 и 4 наибольший общий префикс больше k , поэтому максимальное количество строк, из которых может состоять k -беспрефиксный код - 3.

Во втором примере у любой пары подстрок наибольший общий префикс не больше 2, однако, так как код не может содержать одинаковые строки, больше 3 строк в него не включить.

Решение:

Будем через $\text{lcp}(a, b)$ обозначать наибольший общий префикс (largest common prefix) строк a и b .

Отсортируем массив строк s лексикографически по возрастанию, получим новый массив t . Теперь заметим следующее свойство: для любых $i < j < k$, $\text{lcp}(t[i], t[j]) < \text{lcp}(t[i], t[k])$, то есть чем ближе строки находятся друг к другу в массиве, тем больше у них lcp (иначе был бы нарушен порядок сортировки).

Таким образом, чтобы проверить, что для каких-то $i_1 < i_2 < \dots < i_m$ множество строк $t_{i_1}, t_{i_2}, \dots, t_{i_m}$ является k -беспрефиксным, достаточно проверить, что $\text{lcp}(t_{i_1}, t_{i_2}) \leq k, \text{lcp}(t_{i_2}, t_{i_3}) \leq k, \dots, \text{lcp}(t_{i_{m-1}}, t_{i_m}) \leq k$.

Также несложно заметить, что в k -беспрефиксном коде, состоящем из наибольшего количества строк, первый элемент всегда t_1 , потому что если это не так, первый элемент кода можно заменить на t_1 , и код все еще будет k -беспрефиксным.

Из всего вышесказанного можно вывести следующее решение нашей задачи:

Добавим t_1 в ответ.

Найдем минимальный индекс $i > 1$: $\text{lcp}(t_1, t_i) \leq k$, добавим t_i в ответ

Найдем минимальный индекс $j > i$: $\text{lcp}(t_i, t_j) \leq k$, добавим t_j в ответ

...

Если в очередной раз нужный индекс не нашелся, алгоритм заканчивается, то, что мы успели добавить в ответ, и будет ответом на всю задачу.