

МУНИЦИПАЛЬНЫЙ ЭТАП ВСЕРОССИЙСКОЙ ОЛИМПИАДЫ ШКОЛЬНИКОВ ПО ИНФОРМАТИКЕ В 2024-2025 УЧЕБНОМ ГОДУ ДЛЯ УЧАЩИХСЯ 9-11 КЛАССОВ

Время проведения - 240 минут (4 часа)

Максимальное количество баллов за задачу – 120

Максимальное количество баллов – 600

Особенности проведения - задания практического тура выполняются на компьютерах и сохраняются в отведенные папки. Не допускается использование локальной сети. Для проверки программ используется автоматическая тестирующая система Яндекс.Контест. Интернет-фильтр должен быть настроен только на использование соответствующих адресов и портов. Для авторизации должны быть подготовлены пароли и логины.

Проверяющая система:

<https://official.contest.yandex.ru/contest/69549/enter>

Ограничение по времени – 1 секунда

Ограничение по памяти – 64Mb

Ввод данных – через стандартный поток ввода или из файла input.txt

Вывод данных – через стандартный поток вывода или в файл output.txt

Задача 1. Таинственный дом

На каникулах Петра, Соню, Эдуарда и их младшую сестру Люду отправили в деревню к их дальнему родственнику, профессору Колмогорову. Четверо обычных детей оказались в необыкновенном доме, полном волшебства и таинственных загадок. В каждой комнате их ждало приключение.

Комнат в том доме – просто огромное количество, так что и сам хозяин бывал не во всех. Все комнаты пронумерованы, и если тебе хотелось в комнату с нужным номером, нужно было еще определить на каком она этаже. Дополнительной сложностью является то, что количество комнат на этаже чередуется: на первом этаже a комнат, на втором – b комнат, на третьем – снова a и т.д.

Помогите ребятам определить на каком этаже комната с нужным номером.

Входные данные

На единственной строке расположено через пробел 3 целых числа: $1 \leq a \leq 1000$, $1 \leq b \leq 1000$, $1 \leq k \leq 10^9$ – число комнат на этажах с нечетными номерами, число комнат на этажах с четными номерами и номер нужной комнаты соответственно.

Выходные данные

Необходимо вывести одно целое число – номер этажа, на котором находится комната с номером k . И комнаты, и этажи нумеруются с единицы.

Примеры

№	Входные данные	Выходные данные
1	3 4 8	3
2	3 3 6	2

Решение:

Программа на Python

```
a, b, k = map(int, input().split())
c = k // (a+b)
p = k % (a+b)
n = 0
if p==0:
    n = c*2
elif p<=a:
    n = c*2+1
else:
    n = c*2+2
print(n)
```

Задача 2. Праздник в волшебной стране

Зайдя в одну из комнат, наши герои попали в волшебную страну, где как раз близился большой праздник. Сюда были приглашены все жители – ежи, семья кроликов, фавны, кентавры и многие другие.

Конечно же ребята включились в приготовления! Петр как самый старший взял на себя роль организатора, Соня и Эдуард разносили жителям страны печенье с пригласительными открытками, а маленькая Люда осталась помогать готовить праздничное угощение, и ей очень нужна Ваша помощь.

Люде нужно рассчитать сколько приготовить угощений. Ведь если приготовить слишком мало какого-то блюда, то кто-то из гостей может расстроиться. К счастью, известно сколько и какого блюда съедает каждый из гостей.

Входные данные

На первой строке записаны через пробел два целых числа – общее количество гостей n ($1 \leq n \leq 100$) и количество блюд m ($1 \leq m \leq 1000$). На второй строке через пробел записаны m целых чисел в диапазоне от 1 до 10^9 . Первое число обозначает сколько грамм первого блюда планируется приготовить, второе – сколько будет грамм второго блюда и так далее (для удобства Люда занумеровала все блюда начиная с номера 1).

Далее расположены n строк. На строке с номером i записано через пробел m целых чисел в диапазоне от 1 до 10^9 – сколько грамм каждого блюда съест гость с номером i (первое число соответствует первому блюду, второе – следующему и т.д.).

Выходные данные

Необходимо вывести через пробел в порядке возрастания номера тех блюд, которых не хватит для того, чтобы накормить всех гостей. Если всем всего хватает, нужно вывести единственное число 0.

Примеры

№	Входные данные	Выходные данные
1	3 4 10 20 30 40 1 1 1 1 2 2 2 2 3 18 27 45	2 4
2	2 2 100 100 1 1 2 2	0

Решение:

Из общего количества грамм для каждого блюда последовательно вычитаем вес, который съедает очередной гость. Если в итоге получается отрицательное число, значит данного блюда не хватает.

Программа на Python

```
n, m = map(int, input().split())
a = list(map(int, input().split()))
for i in range(n):
    b = list(map(int, input().split()))
    for j in range(m):
        a[j] -= b[j]
zero = True
for i in range(m):
    if a[i] < 0:
        print(i+1, end=' ')
        zero = False
if zero:
    print(0)
```

Задача 3. Чердак

Конечно же, в таинственном доме профессора Колмогорова есть чердак, и к нему ведет длинная темная лестница. Наши герои очень смелые, но все равно им немного страшно. Поэтому им хочется узнать – сколько имеется способов добраться до чердака по лестнице.

Ребенок начинает с того что встает на первую ступеньку лестницы. При этом он может с текущей ступеньки прыгнуть вверх на любое количество ступенек, кратное k , но не превосходящее m .

Например, если $k = 2$ и $m = 5$, то ребенок с 1й ступеньки может прыгнуть на 3ю (то есть прыгнуть на 2 ступеньки вверх) или на 5ю (то есть прыгнуть на 4 ступеньки вверх). При этом прыгнуть на 7ю ступеньку (то есть на 6 ступенек вверх) ребенок не сможет, так как $6 > 5$.

Если ребенок прыгнул на последнюю ступеньку лестницы или если своим последним прыжком он перепрыгнет эту ступеньку, значит он добрался до чердака. При

этом, если у ребенка получается перепрыгнуть последнюю ступеньку прыжками разной длины, он считает это разными способами добраться до чердака. Например, если он стоит на 17-й ступеньке, а всего лестница состоит из 18 ступеней, и он может сделать прыжки длиной 2 и 4 ступеньки, то у него есть два способа добраться с 17 ступеньки до чердака.

Входные данные

На единственной строке расположены через пробел три целых числа: количество ступенек лестницы $2 \leq n \leq 1000$, а также $1 \leq k \leq 1000$ и m ($1 \leq m \leq n$). При этом можно прыгать вверх на любое количество ступенек, кратное k , но не превосходящее m .

Выходные данные

Вывести единственное целое число – количество способов добраться по лестнице до чердака.

Примеры

№	Входные данные	Выходные данные
1	5 2 5	3
2	3 2 6	3

Пояснения:

В первом случае ребенок может прыгнуть с 1-й ступеньки на 3-ю. Далее он сможет совершить прыжок длиной 2 ступеньки (и попасть на 5-ю ступеньку) или длиной 4 ступеньки (и перепрыгнуть 5-ю ступеньку). Также он может прыгнуть с 1-й сразу на 5-ю ступеньку.

Во втором примере ребенок с первой ступеньки может сделать 3 разных прыжка: на 2 ступеньки (тогда он попадает на последнюю 3-ю ступеньку лестницы), на 4 или 6 ступенек (тогда он перепрыгивает через последнюю ступеньку лестницы 2 разными способами).

Решение:

Задача на динамику. Очередная ячейка обозначает число способов добраться до соответствующей ступеньки. Суммируем значения элементов динамики, отстоящих на расстояние k , $2k$ и т.д. Для того, чтобы получить ответ, нужно учесть ситуации, когда ребенок перепрыгивает через последнюю ступеньку.

Программа на Python

```
n, k, m = map(int, input().split())

f = m // k
delta = f*k

d = [0]*(n+1)
d[1] = 1
for i in range(2, n+1):
    p = i
```

```

while p>0 and i-p<=m:
    d[i]+=d[p]
    p-=k
s = d[n]
for i in range(n-1,max(n-m,0), -1):
    t = (n-i) // k
    s += d[i]*(f-t)
print(s)

```

Задача 4. Магическая книга

В одной из комнат дома Соня нашла страницы древней книги, разбросанные по полу. Книга написана на неизвестном языке. Прочитать эту книгу и узнать её тайны сможет только тот, кто соберёт книгу, сложив страницы по порядку. Сделать это будет непросто, ведь, номера страниц тоже указаны на неизвестном языке. Кто-то уже пробовал, поэтому книга частично собрана: страницы собраны в блоки, все с различным количеством страниц.

Книга волшебная, её страницы вместе удерживаются не с помощью клея, а с помощью магии: две части книги соединяются если их просто приложить друг к другу. За один раз можно соединять только два блока.

Проведя некоторое время с книгой, Соня заметила, что, если соединить два блока страниц в неправильном порядке, они распадаются спустя пару минут. Магия работает только если соединять два самых тонких из имеющихся на данный момент блока. При этом слева должен быть блок с меньшим количеством страниц.

Помогите Соне собрать загадочную книгу.

Входные данные

На первой строке записано одно натуральное число $2 \leq n \leq 10000$ – количество блоков страниц, лежащих на полу. На каждой из следующих n строк указана пара натуральных чисел m и k .

Первое число в строке $1 \leq m \leq 10000$ – номер блока страниц, второе число $2 \leq k \leq 10^{10}$ – количество страниц в блоке.

Выходные данные

Последовательность натуральных чисел, разделённых пробелом: номера блоков в том порядке, в котором они должны стоять в окончательно собранной книге. Гарантируется, что в процессе правильной сборки никакие два блока из имеющихся не совпадают по размеру.

Примеры:

№	Входные данные	Выходные данные
1	2 1 3 2 4	1 2
2	3 1 10 2 15	3 1 2

	3 20	
3	5 1 65 2 200 3 150 4 31 5 43	2 1 4 5 3

Решение:

Задача на применение структур данных. Используется очередь с приоритетом. На каждом шаге соединяем два самых меньших по размеру блока и помещаем обратно в очередь.

Программа на Python

```
from queue import PriorityQueue

n = int(input())
A = []
for i in range(n):
    A.append(tuple(map(int, input().split())))

q = PriorityQueue()
for i in range(n):
    q.put((A[i][1], str(A[i][0])))

while q.qsize() > 1:
    a = q.get()
    b = q.get()
    q.put((a[0]+b[0], a[1] + ' ' + b[1]))

print(q.get()[1])
```

Задача 5. Прятки

Во что играть детям в большом и загадочном доме? Конечно, в прятки! Ребятам уже надоели обычные прятки, поэтому выдумщик Эдуард предложил сыграть в «прятки с запретами». Каждый запрет представляет собой правило: нельзя из комнаты с номером i перейти непосредственно в комнату с номером j , остальные перемещения из комнаты в комнату, не попавшие в запреты, разрешаются.

Все ребята спрячутся по комнатам, затем водящий выходит из своей комнаты и перемещается в любую разрешенную (в том числе уже посещенную до этого), затем из нее в следующую и так далее. Если водящий зашел в комнату, где находится кто-то из других детей, считается что водящий нашел этого ребенка.

Чтобы игра не затягивалась, нужно определить – а всех ли детей можно найти следуя таким правилам?

Входные данные

На первой строке записаны три целых числа: $1 \leq n \leq 100$, $n \leq m \leq 100$ и $1 \leq k \leq 1000$ – количество детей, участвующих в игре, количество комнат в доме и количество запретов соответственно.

На каждой из следующих n строк располагается, во-первых, имя ребенка – последовательность символов, состоящая из прописных и строчных букв латинского алфавита, количество символов в имени не более 10. Во-вторых, через пробел от имени ребенка записан номер комнаты (целое число в диапазоне от 1 до m), в которой этот ребенок находится. Все имена детей различны, комнаты нумеруются с 1, в каждой комнате может находиться не более одного ребенка.

На следующей строке находится имя водящего. Это – одно из имен, записанных на предыдущих n строках.

На каждой из следующих k строк записаны через пробел два различных целых числа $1 \leq a \leq m$ и $1 \leq b \leq m$ – эта запись соответствует правилу: нельзя из комнаты с номером a перемещаться непосредственно в комнату с номером b (при этом перемещение в обратном направлении из комнаты с номером b в комнату с номером a правилами разрешено, если нет соответствующего запрета). Запреты не дублируются.

Выходные данные

Необходимо вывести в лексикографическом порядке (как в словаре) имена детей, которых водящий рано или поздно сможет найти, по одному на строку. Если не получится найти ни одного ребенка нужно вывести фразу “Bad karma” без кавычек.

Примеры

№	Входные данные	Выходные данные
1	4 4 1 Petr 1 Sonya 2 Eduard 4 Luda 3 Petr 1 3	Eduard Luda Sonya
2	3 4 3 Petr 1 Sonya 2 Luda 3 Petr 1 2 3 2 3 4	Luda Sonya
3	3 3 2 Petr 1 Sonya 2 Luda 3 Sonya 2 1	Bad karma

№	Входные данные	Выходные данные
1	4 4 1 Petr 1 Sonya 2 Eduard 4 Luda 3 Petr 1 3	Eduard Luda Sonya
	2 3	

Пояснения:

В первом примере запрещено из первой комнаты двигаться сразу в 3ю. Но водящий Петр может перейти из 1й во вторую, из второй в четвертую, и уже из четвертой в третью. Таким образом он найдет всех трех спрятавшихся ребят.

Во втором примере запрещено перемещаться во вторую комнату из первой и третьей, а также из третьей в четвертую. Петр может, например, из первой перейти в третью, дальше вернуться в первую, перейти в четвертую (так как всего комнат четыре) и уже оттуда попасть во вторую. Таким образом, он найдет всех ребят.

В третьем примере водит Соня, которая не может из второй попасть ни в первую, ни в третью. Она никого не сможет найти.

Решение:

Задача на графы. Необходимо выделить компоненту связности, в которой находится Водящий. В ответ выводим имена всех детей, которые находятся в этой компоненте.

Программа на Python

```
n, m, k = map(int, input().split())
child = {}
rooms = {}
for i in range(n):
    s, num = input().split()
    t = int(num)-1
    child[s] = t
    rooms[t] = s

voda = input()
start = child[voda]

g = []
for i in range(m):
    g.append([True]*m)

for i in range(k):
    a, b = map(int, input().split())
    g[a-1][b-1] = False

used = [False]*m
def dfs(v):
    used[v] = True
```



```
        for i in range(m):
            if g[v][i] and not used[i]:
                dfs(i)
dfs(start)

res = []
for i in range(m):
    if used[i] and i!=start and i in rooms:
        res.append(rooms[i])
if len(res)>0:
    res.sort()
    for r in res:
        print(r)
else:
    print("Bad karma")
```