

Задача А. Средняя зарплата

Имя входного файла: a.in
 Имя выходного файла: a.out
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

В фирме MacroHard работают N сотрудников, каждый из которых получает зарплату, выражающуюся целым числом рублей. Известно, что ни один сотрудник не получает меньше 5000 рублей, и никто не получает больше 100000 рублей. Также известно, что средняя зарплата сотрудника в этой фирме выражается целым числом копеек и составляет A рублей B копеек.

Журналист, готовя публикацию об этой фирме, решил привести зарплаты всех сотрудников. Однако оказалось, что это коммерческая тайна. Журналиста это не смутило, и он решил придумать всем сотрудникам зарплаты. Однако у него возникла сложность – для правдоподобности должны выполняться все общеизвестные ограничения (зарплаты должны выражаться целым числом рублей из диапазона от 5000 до 100000, и вычисление средней зарплаты должно в точности приводить к результату A рублей B копеек).

Помогите ему! Напишите программу, которая по введенным числам N , A , B «придумает» и выведет N зарплат. Гарантируется, что решение существует.

Формат входных данных

Вводятся натуральное число N ($1 \leq N \leq 100$), натуральное число A ($10000 \leq A \leq 30000$) и целое число B ($0 \leq B \leq 99$).

Формат выходных данных

Выведите N целых чисел, выражающих зарплаты сотрудников в рублях. Если возможных вариантов распределения зарплат несколько, выведите любой из них.

Примеры

a.in	a.out
5 10000 0	10000 10000 10000 10000 10000
2 20000 50	20000 20001

Задача В. Лягушка

Имя входного файла: b.in
 Имя выходного файла: b.out
 Максимальное время работы на одном тесте: 0.4 секунды
 Максимальный объем используемой памяти: 64 мегабайта

N -лягушка живет на болоте, на котором в ряд растут бесконечно много кувшинок, пронумерованных слева направо числами 1, 2, 3, ...

Изначально N -лягушка сидит на кувшинке с номером K ($K > N$). Каждый раз N -лягушка прыгает на N кувшинок влево и повторяет это, пока не оказывается на номере, меньше либо равном N . Если она попадает на кувшинку с номером N , то становится счастливой, и дальше никуда не прыгает. Если же она попадает на кувшинку с каким-нибудь номером $M < N$, то огорчается, прыгает на N кувшинок вправо и превращается в M -лягушку (теперь она будет прыгать на M клеток влево и мечтать попасть на клетку номер M , а если у нее это не получится, то она превратится в X -лягушку, и так далее).

Требуется выяснить, исполнятся ли когда-либо мечты N -лягушки, сидящей изначально на кувшинке с номером K , и если да, то на какой кувшинке она окажется.

Формат входных данных

Вводятся два натуральных числа N и K . $1 \leq N < K \leq 2 \cdot 10^9$.

Формат выходных данных

Выведите номер кувшинки, на которой останется N -лягушка. Если мечты лягушки никогда не исполнятся, выведите одно число 0.

Примеры

b.in	b.out
2 10	2
2 5	1

Задача С. Кодовый замок

Имя входного файла: c.in
 Имя выходного файла: c.out
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

Кодовый замок состоит из N рычажков, каждый из которых может быть установлен в любое из K положений, обозначенных натуральными числами от 1 до K . Известно, что для того чтобы открыть замок, нужно, чтобы сумма положений любых трех последовательных рычажков была равна K .

Два рычажка уже установлены в некоторые положения, и их переключать нельзя. Рычажок с номером p_1 установлен в положение v_1 , а рычажок p_2 – в положение v_2 .

Напишите программу, которая определит, сколькими способами можно установить остальные рычажки, чтобы открыть замок.

Формат входных данных

Вводятся натуральные числа N , K , p_1 , v_1 , p_2 , v_2 . Рычажки пронумерованы числами от 1 до N .

$$3 \leq N \leq 10000, 3 \leq K \leq 6, p_1 \neq p_2, 1 \leq p_1 \leq N, 1 \leq p_2 \leq N, 1 \leq v_1 \leq K, 1 \leq v_2 \leq K.$$

Формат выходных данных

Выведите одно число — количество искомых комбинаций или 0, если, соблюдая все условия, замок открыть невозможно.

Примеры

c.in	c.out
3 3 1 1 2 1	1
3 3 1 1 3 2	0
4 4 1 1 4 1	2
5 3 1 1 4 1	1

Задача D. Смайлики

Имя входного файла: d.in
 Имя выходного файла: d.out
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

Напишите программу, которая посчитает количество смайликов в заданном тексте.

Смайликом будем считать последовательность символов, удовлетворяющую условиям:

- первым символом является либо ; (точка с запятой) либо : (двоеточие) ровно один раз
- далее может идти символ - (минус) сколько угодно раз (в том числе символ минус может идти ноль раз)
- в конце обязательно идет некоторое количество (не меньше одной) одинаковых скобок из следующего набора: (,), [,].
- внутри смайлика не может встречаться никаких других символов.

Например, нижеприведенные последовательности являются смайликами:

```

        :)
        ;-----[[[[[[[[
    
```

в то время как эти последовательности смайликами не являются (хотя некоторые из них содержат смайлики):

```

        :-) ]
        ;--
        -)
        ::-(
        :-( )
    
```

В этой задаче надо будет посчитать количество смайликов, содержащихся в данном тексте.

Формат входных данных

Вводится одна строка текста, которая может содержать маленькие латинские буквы, пробелы, символы, которые могут встречаться в смайликах. Длина строки не превышает 200 символов.

Формат выходных данных

Выведите одно число — количество смайликов, которые встречаются в тексте.

Примеры

d.in	d.out
:) ;-----[[[[[[[[2
:-)] ;----;	1
-) (----:----	0
hello :-)	1

Задача Е. Субботник

Имя входного файла: e.in
 Имя выходного файла: e.out
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

В классе учатся N человек. Классный руководитель получил указание разбить их на R бригад по C человек в каждой и направить на субботник ($N = R \cdot C$).

Все бригады на субботнике будут заниматься переноской бревен. Каждое бревно одновременно несут все члены одной бригады. При этом бревно нести тем удобнее, чем менее различается рост членов этой бригады.

Числом неудобства бригады будем называть разность между ростом самого высокого и ростом самого низкого членов этой бригады (если в бригаде только один человек, то эта разница равна 0). Классный руководитель решил сформировать бригады так, чтобы максимальное из чисел неудобства сформированных бригад было минимально. Помогите ему в этом!

Рассмотрим следующий пример. Пусть в классе 8 человек, рост которых в сантиметрах равен 170, 205, 225, 190, 260, 130, 225, 160, и необходимо сформировать две бригады по четыре человека в каждой. Тогда одним из вариантов является такой:

1 бригада: люди с ростом 225, 205, 225, 260

2 бригада: люди с ростом 160, 190, 170, 130

При этом максимальное число неудобства будет во второй бригаде, оно будет равно 60, и это наилучший возможный результат.

Формат входных данных

Сначала вводятся натуральные числа R и C — количество бригад и количество человек в каждой бригаде ($1 \leq R \cdot C \leq 1000$). Далее вводятся $N = R \cdot C$ целых чисел — рост каждого из N учеников. Рост ученика — натуральное число, не превышающее 1 000 000 000.

Формат выходных данных

Выведите одно число — наименьшее возможное значение максимального числа неудобства сформированных бригад.

Примеры

e.in	e.out
2 4 170 205 225 190 260 130 225 160	60

Задача F. Язык Мумба-Юмба

Имя входного файла: `f.in`
 Имя выходного файла: `f.out`
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

Слова в языке Мумба-Юмба могут состоять только из букв **a**, **b**, **c**, и при этом:

- никогда не содержат двух букв **b** подряд,
- ни в одном слове никогда не встречается три одинаковых подслова подряд. Например, по этому правилу в язык Мумба-Юмба не могут входить слова **aaa** (так как три раза подряд содержит подслово **a**), **ababab** (так как три раза подряд содержит подслово **ab**), **aabcabcsabca** (три раза подряд содержит подслово **abc**).

Все слова, удовлетворяющие вышеописанным правилам, входят в язык Мумба-Юмба.

Напишите программу, которая по данному слову определит, принадлежит ли оно этому языку.

Формат входных данных

Вводится одно слово, состоящее только из строчных букв **a**, **b**, **c**, длины не более 100.

Формат выходных данных

Если слово входит в язык Мумба-Юмба, выведите YES, в противном случае выведите NO.

Примеры

<code>f.in</code>	<code>f.out</code>
abca	YES
abcabccaaa	NO

Задача G. Операционные системы

Имя входного файла: g.in
 Имя выходного файла: g.out
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

Васин жесткий диск состоит из M секторов. Вася последовательно устанавливал на него различные операционные системы следующим методом: он создавал новый раздел диска из последовательных секторов, начиная с сектора номер a_i и до сектора b_i включительно, и устанавливал на него очередную систему. При этом, если очередной раздел хотя бы по одному сектору пересекается с каким-то ранее созданным разделом, то ранее созданный раздел «затирается», и операционная система, которая на него была установлена, больше не может быть загружена.

Напишите программу, которая по информации о том, какие разделы на диске создавал Вася, определит, сколько в итоге работоспособных операционных систем установлено и работает в настоящий момент на Васином компьютере.

Формат входных данных

Сначала вводятся натуральное число M — количество секторов на жестком диске ($1 \leq M \leq 10^9$) и целое число N — количество разделов, которое последовательно создавал Вася ($0 \leq N \leq 1000$).

Далее идут N пар чисел a_i и b_i , задающих номера начального и конечного секторов раздела ($1 \leq a_i \leq b_i \leq M$).

Формат выходных данных

Выведите одно число — количество работающих операционных систем на Васином компьютере.

Примеры

g.in	g.out
10 3 1 3 4 7 3 4	1
10 4 1 3 4 5 7 8 4 6	3

Задача Н. Пицца

Имя входного файла: h.in
 Имя выходного файла: h.out
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

Вы решили заказать пиццу с доставкой на дом. Известно, что для клиентов, сделавших заказ на сумму более C рублей, доставка является бесплатной, при заказе на C рублей и меньше доставка стоит B рублей.

Вы уже выбрали товара стоимостью A рублей. В наличии имеются еще N товаров стоимостью d_1, \dots, d_N рублей, каждый в единственном экземпляре. Их также можно включить в заказ.

Как потратить меньше всего денег и получить на дом уже выбранный товар стоимостью A рублей?

Формат входных данных

Вводятся сначала числа A, B, C, N , а затем N чисел d_1, \dots, d_N .

Все числа целые, $1 \leq A \leq 1000$, $1 \leq B \leq 1000$, $1 \leq C \leq 1000$, $0 \leq N \leq 1000$, $1 \leq d_i \leq 1\,000\,000$.

Формат выходных данных

Выведите единственное число – суммарное количество денег, которое придется потратить.

Примеры

h.in	h.out
10 17 25 5 2 7 5 3 7	26
100 1 50 5 5 2 4 3 1	100
10 14 25 5 2 7 5 3 7	24

Пояснение

В первом примере экономнее всего докупить 1, 2 и 5 товары. Во втором ничего докупать не надо, ведь доставка уже стала бесплатной. В третьем дешевле всего заплатить за доставку самому.