

C12A masala. Иррациональная задача

Маленькому Пете задали в качестве домашнего задания следующую задачу:
Дана функция $f(x) = (((x \bmod p_1) \bmod p_2) \bmod p_3) \bmod p_4$ (mod обозначает операцию взятия остатка по модулю). Требуется посчитать количество целых чисел x из промежутка $[a;b]$ таких, что $f(x) = x$.

К сожалению, Петя забыл, в каком порядке должны браться остатки, а просто записал 4 числа. С равной вероятностью может оказаться, что брать остатки для вычисления функции нужно в любом из 24 возможных порядков. К примеру, если у Пети есть числа 1, 2, 3, 4, то он может брать остатки именно в таком порядке, а может сначала брать остаток по модулю 4, затем по модулям 2, 3, 1. Также существует 22 других перестановки этих чисел, в порядке которых могут браться остатки. В этой задаче 4 числа, записанных Петей, будут попарно различны.

Теперь Петя уже не может выполнить задачу так, как от него требовал учитель, но ради развлечения он решил найти количество таких целых чисел $x \in [a; b]$, что вероятность того, что для них $f(x) = x$, не меньше чем 31.4159265352718281828459045%. Иными словами, Пете подойдут те числа x , для которых существует не менее 7 таких перестановок чисел p_1, p_2, p_3, p_4 , что $f(x)=x$.

Входящие данные

В первой строке входного файла находятся 6 целых чисел, разделенных пробелами: p_1, p_2, p_3, p_4, a, b ($1 \leq p_1, p_2, p_3, p_4 \leq 1000, 0 \leq a \leq b \leq 31415$).

Гарантируется, что числа p_1, p_2, p_3, p_4 будут попарно различны.

Выходящие данные

Выведите количество целых чисел из заданного интервала, удовлетворяющих заданным условиям..

Пример

Входящие данные	Выходящие данные
2 7 1 8 2 8	0
20 30 40 50 0 100	20
31 41 59 26 17 43	9

C12B masala. Bit++

Классический язык программирования в Битландии — Bit++. Этот язык очень необычный и сложный!

Особенность языка Bit++ состоит в том, что в нем ровно одна переменная, которая называется x . Также в этом языке есть две операции:

- Операция ++ увеличивает значение, хранящееся в переменной x на 1.
- Операция -- уменьшает значение, хранящееся в переменной x на 1.

Предложение в языке Bit++ — это последовательность, состоящая из ровно одной операции и ровно одной переменной x . Предложение записывается без пробелов, то есть может содержать только символы: «+», «-», «X». Выполнить предложение, значит выполнить операцию, которая содержится в предложении.

Программа на языке Bit++ — это последовательность предложений, каждое из которых требуется выполнить. Выполнить программу, значит выполнить все ее предложения.

Вам задана программа на языке Bit++. Перед выполнением программы в переменной x хранится значение 0. Выполните программу и выведите значение, которое будет храниться в переменной x после выполнения программы.

Входящие данные

В первой строке записано единственное целое число n ($1 \leq n \leq 150$) — количество предложений в программе. В каждой из следующих n строк записано по предложению. Каждое предложение состоит из ровно одной операции (++ или --) и ровно одной переменной x (обозначается буквой «X»). Таким образом, не бывает пустых предложений. Операция и переменная могут быть записаны в любом порядке.

Выходящие данные

Выведите единственное целое число — значение, которое будет храниться в переменной x после выполнения программы.

Примерlar

Входящие данные	Выходящие данные
1 ++X	1
2 X++ --X	0

C12C masala. Exponentiation

Problems involving the computation of exact values of very large magnitude and precision are common. For example, the computation of the national debt is a taxing experience for many computer systems.

This problem requires that you write a program to compute the exact value of R^n where R is a real number ($0.0 < R < 99.999$) and n is an integer such that $0 < n \leq 25$.

Входящие данные

The input will consist of a set of pairs of values for R and n . The R value will occupy columns 1 through 6, and the n value will be in columns 8 and 9.

Выходящие данные

The output will consist of one line for each line of input giving the exact value of R^n . Leading zeros should be suppressed in the output. Insignificant trailing zeros must not be printed. Don't print the decimal point if the result is an integer.

Примерлар

Kiruvchi	Chiquvchi
95.123 12	548815620517731830194541.899025343415715973535967221869852721
0.4321 20	.00000005148554641076956121994511276767154838481760200726351203835429763013462401
5.1234 15	43992025569.928573701266488041146654993318703707511666295476720493953024
6.7592 9	29448126.764121021618164430206909037173276672

C12D masala. Mashina nomeri

Абдулла купил новый автомобиль. Теперь он купил номер для своего автомобиля и он на том месте поинтересовался который номер вышедший оттуда. Помогите вычислить Абдулла это.

Сейчас номер машин в виде 11 А 111 АА. Числа изменяются следующем порядке:

1	1	A	1	1	1	A	A
8	7	4	3	2	1	5	6

Регионы	Номера кодов
Toshkent shahri	01—09
Toshkent viloyati	10—19
Sirdaryo viloyati	20—24
Jizzax viloyati	25—29
Samarqand viloyati	30—39
Farg'ona viloyati	40—49
Namangan viloyati	50—59
Andijon viloyati	60—69
Qashqadaryo viloyati	70—74
Surxondaryo viloyati	75—79
Buxoro viloyati	80—84
Navoiy viloyati	85—89
Xorazm viloyati	90—94
Qoraqalpog'iston Respublikasi	95—99

Входящие данные

В одном строке дается номер машины

Выходящие данные

На первом строке вывести на какую региону относится машина, на втором строке вывести который номер на этом регионе.

Входящие данные	Выходящие данные
01A000AA	Toshkent shahri 1
50C606SS	Namangan viloyati 12638607
11I111II	Toshkent viloyati 23200112

C12E masala. Несчастливый билет

Наверное, каждый когда-нибудь ездил в общественном транспорте и оплачивал проезд. Получив свой билет (традиционно в билете четное количество цифр), многие проверяют, является ли этот билет счастливым. Напомним, что билет называется счастливым, если сумма цифр в его первой половине совпадает с суммой цифр во второй.

Но, разумеется, далеко не каждый билет счастливый. Более того, иногда, посмотрев на билет, можно сразу сказать, что счастливым он не является. Поэтому рассмотрим следующий критерий несчастья, по которому точно можно определить несчастливый билет. Будем говорить, что билет точно не счастливый, если каждой цифре его первой половины можно сопоставить некоторую цифру второй половины так, что каждая цифра из первой половины строго меньше сопоставленной цифры из второй половины, либо каждая цифра из первой половины строго больше сопоставленной цифры из второй половины. Каждую цифру в сравнениях нужно использовать в точности один раз. Другими словами, найдется такое взаимно-однозначное соответствие между цифрами первой и второй половины билета, что либо каждая цифра первой половины окажется строго меньше соответствующей ей цифры второй половины, либо каждая цифра первой половины окажется строго больше соответствующей ей цифры второй половины.

Например, билет 2421 удовлетворяет данному критерию несчастья и счастливым являться не будет (искомое соответствие $2 > 1$ и $4 > 2$), билет 0135 также удовлетворяет критерию (искомое соответствие $0 < 3$ и $1 < 5$), а билет 3754 критерию не удовлетворяет.

У Вас в руках находится билет, состоящий из $2n$ цифр. А Ваша задача — проверить, выполняется ли для него критерий несчастья.

Входящие данные

В первой строке задано целое число n ($1 \leq n \leq 100$). Во второй строке содержится строка, состоящая из $2n$ цифр, задающая Ваш билет.

Выходящие данные

В первую строку выведите «YES», если билет удовлетворяет критерию несчастья, и «NO» в противном случае (без кавычек).

Входящие данные	Выходящие данные
2 2421	YES
2 0135	YES
2 3754	NO