



Twoscomp: Дополнительный код

Имя входного файла: twoscomp.in
Имя выходного файла: twoscomp.out

Дано целое число A и натуральное n . Выведите запись числа A в двоичном n -разрядном дополнительном коде.

Формат входного файла

Программа получает на вход числа A и n : $2 \leq n \leq 16$, $-2^{n-1} \leq A \leq 2^{n-1} - 1$.

Формат выходного файла

Программа должна вывести последовательность из n нулей и единиц.

Примеры

twoscomp.in	twoscomp.out
5 8	00000101
-5 8	11111011

Binfrac: Записать двоичную дробь

Имя входного файла: binfrac.in
Имя выходного файла: binfrac.out

Запишите данное действительное число приближенно в двоичной системе счисления в виде дробного числа с фиксированной точкой.

Формат входного файла

Программа получает на вход действительное неотрицательное число, не превосходящее 100, записанное в десятичной системе счисления с фиксированной точкой.

Формат выходного файла

Программа должна вывести представление этого числа в двоичной системе счисления с фиксированной точкой. Ответ будет проверяться с абсолютной погрешностью 2^{-32} , поэтому достаточно вывести 32 значащие двоичные цифры после точки.

Примеры

binfrac.in	binfrac.out
3.25	11.01
4	100
0.1	0.0001100110011001100110011001100110011001100110011

Три следующие задачи даны в двух вариантах: extended и double. Задачи должны решаться при помощи операций со стандартными действительными типами данных (описанных в IEEE-754). Использование нестандартной арифметики (например, типа данных decimal) запрещено. Варианты отличаются ограничениями на входные данные. Вариант “extended” предполагает использование чисел расширенной точности (тип данных long double в языке C++), и эти задачи можно сдавать только на языке C++ (при этом могут быть проблемы при использовании Visual C++). Вариант “double” допускает решение с использованием действительных чисел двойной точности и в этом варианте можно использовать только языки Python или Java.

Diet: Диета

Имя входного файла: diet.in
Имя выходного файла: diet.out

В некоторой сверхсекретной лаборатории изучаются физические возможности животных. Любой живой организм нуждается в белках, жирах и углеводах. Известен набор продуктов, имеющийся в распоряжении лаборатории, и меню животных — сколько единиц каждого продукта они получают. Известно также, сколько белков, жиров и углеводов необходимо для нормальной жизнедеятельности животного. Необходимо определить, получает ли животное достаточное количество питательных веществ.

Известно, что животному требуется в сутки X белков, Y жиров и Z углеводов. Известно также, что всего животное получает в сутки N продуктов питания, и для каждого из них известны A_i , B_i , C_i и Q_i — энергетическая ценность единицы продукта в белках, жирах и углеводах и количество единиц этого продукта.

Формат входного файла

На первой строке входных данных записаны действительные числа X , Y и Z . На второй строке записано целое число N ($0 \leq N \leq 25000$). Далее на N строках записаны действительные числа A_i , B_i , C_i и Q_i . Все числа X , Y , Z , A_i , B_i , C_i , Q_i заданы с точностью до 5 знаков после запятой и не превосходят 100 001.

Для тех, кто пишет на Python или Java числа X , Y , Z не превосходят 25000, числа A_i , B_i , C_i , Q_i не превосходят 100.

Формат выходного файла

Выведите YES, если данный пищевой рацион является достаточным по всем параметрам, и NO в противном случае.



Примеры

diet.in	diet.out
1.0 1.0 1.0 3 1 0 0 1 0 0.5 0 2 0 0 0.25 4	YES

Online-2: Точки на прямой

Имя входного файла: `online2.in`
Имя выходного файла: `online2.out`

Даны три точки в трехмерном пространстве. Проверьте, лежат ли они на одной прямой.

Формат входного файла

Программа получает на вход координаты трех точек $x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3$. Координаты являются действительными числами, не превосходящими 10^6 по модулю, заданными с точностью до 3 знаков после точки.

Для тех, кто пишет на Python или Java координаты не превосходят 10^3 и заданы с точностью до 3 знаков после точки.

Формат выходного файла

Выведите YES, если точки лежат на одной прямой, или NO в противном случае.

Примеры

online2.in	online2.out
999999.000 -999999.000 999999.000 999999.001 -999999.002 999999.003 999999.002 -999999.004 999999.006	YES
999999.000 -999999.000 999999.000 999999.001 -999999.002 999999.003 999999.002 -999999.004 999999.007	NO

Rectangle: Прямоугольник

Имя входного файла: `rectangle.in`
Имя выходного файла: `rectangle.out`

На плоскости задан прямоугольник (стороны которого не обязательно параллельны осям координат). Даны координаты трех его вершин, найдите его четвертую вершину.

Формат входного файла

Программа получает на вход координаты трех вершин прямоугольника $x_1, y_1, x_2, y_2, x_3, y_3$. Координаты: действительные числа, не превосходящие 10^7 по модулю и имеющие не более 9 десятичных цифр после точки.

Координаты заданы абсолютно точно. Порядок записи вершин произвольный. Для тех, кто пишет на Python или Java координаты не превосходят 10^5 и имеют не более 9 десятичных цифр после точки.

Формат выходного файла

Выведите координаты четвертой вершины прямоугольника. Ответ должен быть получен точно (допускается не выводить незначащие нули). В частности, это означает, что если в правильном ответе должно быть число 1.0, то ответ 0.9999999999999992 неправильный, а ответ 1.0000000000000000 — правильный.

Примеры

rectangle.in	rectangle.out
-1 2 1 1 2.6 4.2	0.6 5.2

Epsilon: Машинное эpsilon

Имя входного файла: `epsilon.in`
Имя выходного файла: `epsilon.out`

Эту задачу можно сдавать только на языке C++, т.к. необходимо использовать три типа данных: float, double, long double.

Напомним, что машинным эpsilonом для некоторого типа называется такое наименьшее положительное число ϵ , представляемое в данном типе, что $1 + \epsilon \neq 1$ при вычислении в данном типе. Значение машинного эpsilon зависит от типа данных, используемого для представления действительных чисел.

Напишите программу, которая вычисляет машинное эpsilon для типов данных одинарной, двойной и расширенной точности. Решение не должно использовать какие-либо специальные знания о формате хранения действительного числа (например, о размере мантиссы числа).

Формат входного файла

Входные данные в этой задаче отсутствуют.

Формат выходного файла

Программа выводит три действительных числа в формате с плавающей точкой: значения машинного эpsilon для чисел одинарной, двойной и расширенной точности.

Проверка будет осуществляться путем сравнением ответов с правильными с относительной погрешностью 10^{-6} .