1. Возрастающая последовательность

ограничение по времени на тест

1 second

ограничение по памяти на тест

64 megabytes

ввод

стандартный ввод

вывод

стандартный вывод

Последовательность *a*0, *a*1, ..., *at*- 1 называется возрастающей если *ai*- 1 < *ai* для всех *i*: 0 < *i* < *t*.

Вам задана последовательность *b*0, *b*1, ..., *bn*- 1 и натуральное число *d*. Каждый ход выбирается один из элементов последовательности и увеличивается на *d*. Какое минимальное число ходов необходимо совершить, чтобы сделать последовательность возрастающей?

**Входные данные**

Первая строка входных данных содержит два натуральных числа *n* и *d* (2 ≤ *n* ≤ 2000;1 ≤ *d* ≤ 106). Вторая строка содержит элементы последовательности *b*0, *b*1, ..., *bn*- 1 (1 ≤ *bi* ≤ 106). Числа в строках разделяются пробелами.

**Выходные данные**

Выведите искомое наименьшее количество ходов.

**Примеры**

**входные данные**

4 2  
1 3 3 2

**выходные данные**

3

2. Джек-попрыгунчик

ограничение по времени на тест

1 second

ограничение по памяти на тест

64 megabytes

ввод

стандартный ввод

вывод

стандартный вывод

В настоящее время Джек работает над развитием своих навыков прыгать в длину. Сейчас он находится в точке 0 координатной прямой. Его цель — оказаться в точке *x*. В целях тренировки его первый прыжок будет иметь длину ровно 1, второй — ровно 2 и т.д. Каждый раз он выбирает прыгнуть ему влево или вправо.

Какое минимальное количество прыжков должен совершить Джек, чтобы попасть в точку *x*.

**Входные данные**

Входные данные содержат целое число *x* ( - 109 ≤ *x* ≤ 109).

**Выходные данные**

Выведите наименьшее количество прыжков, необходимых для попадания в *x*.

**Примеры**

**входные данные**

2

**выходные данные**

3

**входные данные**

6

**выходные данные**

3

**входные данные**

0

**выходные данные**

0

3 The Elder Trolls IV: Oblivon

ограничение по времени на тест

2 seconds

ограничение по памяти на тест

256 megabytes

ввод

стандартный ввод

вывод

стандартный вывод

Вася играет в The Elder Trolls IV: Oblivon. Ох уж эти создатели компьютерных игр! Чего только они не придумают! В The Elder Trolls IV: Oblivon появились совершенно уникальные монстры. Одним из них является Неубивамемый Слизень. Почему "неубиваемый"? Во-первых, потому, что убить его можно только режущим оружием, так что любителям двуручных янтарных молотов необходимо подыскивать себе подходящий ножичек. Во-вторых, необходимо нанести Неубиваемому Слизню очень много режущих ударов. Крайне много. Слишком много!

Вася уже успел прокачать своего персонажа до 80 уровня и для получения 81-ого ему было поручено убить Неубиваемого Слизня. Монстр имеет очень интересную форму. Он представляет собой прямоугольный параллелепипед размером *x* × *y* × *z*, состоящий из неразрушаемых клеток, размером 1 × 1 × 1. За один удар персонаж Васи может разрубить Слизня целиком вдоль линии воображаемой сетки, т.е. плоскостью параллельной одной из граней параллелепипеда. Монстр погибает только тогда, когда количество частей, на которые его разрубил Вася, достигает некоторого критического числа.

После каждого удара все части монстра не распадаются, а остаются на своем месте. То есть Вася может разрубать сразу несколько частей одним ударом.

Вася хочет узнать на какое максимальное количество частей он может разрубить Неубиваемого Слизня, нанеся ему не более чем *k* ударов.

Персонаж Васи пользуется абсолютно тонким мечом бесконечной длины.

**Входные данные**

В единственной строке заданы 4 целых числа *x*, *y*, *z*, *k* (1 ≤ *x*, *y*, *z* ≤ 106, 0 ≤ *k* ≤ 109).

**Выходные данные**

Вывести единственное число — ответ на задачу.

Пожалуйста, не используйте спецификатор %lld для чтения или записи 64-х битовых чисел на С++. Рекомендуется использовать поток cout (также вы можете использовать спецификатор %I64d).

**Примеры**

**входные данные**

2 2 2 3

**выходные данные**

8

**входные данные**

2 2 2 1

**выходные данные**

2

**Примечание**

В первом примере Вася проводит 3 попарно перпендикулярных разреза. Первым разрезом он разрубает монстра на две части, затем каждую часть разрубает еще на две вторым разрезом, потом каждую из 4 частей он снова разрубает на две последним разрезом.

4. Расстановка

ограничение по времени на тест

2 seconds

ограничение по памяти на тест

256 megabytes

ввод

стандартный ввод

вывод

стандартный вывод

Вася купил полное собрание сочинений известного берляндского поэта Пети в *n* томах. Тома нумеруются от 1 до *n*. Вася решил поставить их на одну полку, на которой как раз было *n* свободных мест, пронумерованных от 1 до *n*. Он считает, что ставить книги просто по порядку — некрасиво. Вася хочет минимизировать количество делителей расстановки — таких целых положительных чисел *i*, что для хотя бы одного *j* (1 ≤ *j* ≤ *n*) выполняется: *j* *mod* *i* = 0 и в то же время *p*(*j*) *mod* *i* = 0, где *p*(*j*) — номер тома, который стоит на *j*-ом месте, а *mod* — операция взятия остатка от деления. Естественно, один том может занимать ровно одно место, и на одном месте стоит ровно один том.

Помогите Васе — найдите расстановку с наименьшим числом делителей.

**Входные данные**

В первой строке записано число *n* (1 ≤ *n* ≤ 100000) — количество томов и свободных мест.

**Выходные данные**

Выведите *n* чисел — искомую расстановку с наименьшим количеством делителей. *j*-ое число (1 ≤ *j* ≤ *n*) должно быть равно *p*(*j*) — номеру тома, который стоит на *j*-ом месте. Если решений несколько, выведите любое.

**Примеры**

**входные данные**

2

**выходные данные**

2 1

**входные данные**

3

**выходные данные**

1 3 2

*Сумма*

(Память 16Мб Время 1сек)

Требуется сложить два целых числа А и B.

Входные данные:

Во входном файле INPUT.TXT записано два неорицательных целых числа, не превышающих 10^1000, по одному в каждой строке.

Выходные данные:

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число- сумму чисел A и B, безлидирующих нулей.

**Примеры**

**входные данные**

1

2

**выходные данные**

3

**входные данные**

333333333333

44444444444

**выходные данные**

377777777777