

## Содержание

<b>Часть 1</b>	<b>2</b>
1 Задача А. Plus minus [0.5 секунд, 256 mb]	2
2 Задача В. Поиск [1 секунда, 256 mb]	3
3 Задача С. Быстрое прибавление [4 секунды, 256 mb]	4
4 Задача D. Линейная сумма [3 секунды, 256 mb]	5
5 Задача Е. Длинное выражение [2 секунды, 256 mb]	6
<b>Часть 2</b>	<b>7</b>
6 Задача F. Точки и отрезки [0.5 секунд, 256 mb]	7
7 Задача G. Обмен [0.5 секунд, 256 mb]	8

## Часть 1

### 1 Задача А. Plus minus [0.5 секунд, 256 mb]

В каждой клетке поля  $M \times N$  стоит либо плюс, либо минус. За один ход разрешается поменять знаки на противоположные в любом квадрате  $2 \times 2$ . Можно ли с помощью таких операций получить во всех клетках поля знаки плюс?

#### Формат входных данных

В первой строке числа  $M$  и  $N$  ( $1 \leq N, M \leq 1000$ ). В следующих  $M$  строках содержится по  $N$  символов +, либо -.

#### Формат выходных данных

Ответ на вопрос задачи: слово Yes или No

#### Пример

plusminus.in	plusminus.out
3 3 -+- -+- ++-	No
3 3 -+- +++ -+-	Yes

## 2    Задача В. Поиск [1 секунда, 256 mb]

В этой задаче нужно уметь выяснять, содержится ли число в последовательности.

### Формат входных данных

В первой строке входного файла заданы через пробел два целых числа  $n$  и  $k$  ( $1 \leq n \leq 300\,000$ ,  $1 \leq k \leq 300\,000$ ). Во второй строке задана последовательность из  $n$  отсортированных целых чисел  $a_1, a_2, \dots, a_n$ , записанных через пробел ( $1 \leq a_i \leq 10^9$ ). В третьей строке записаны запросы —  $k$  целых чисел  $b_1, b_2, \dots, b_k$  записанных через пробел, в порядке возрастания ( $1 \leq b_j \leq 10^9$ ).

### Формат выходных данных

В выходной файл выведите  $k$  строк. В  $j$ -ой строке выведите “YES”, если число  $b_j$  содержится в последовательности  $\{a_i\}$ , и “NO” в противном случае.

### Примеры

find2.in	find2.out
3 3	NO
2 3 5	YES
1 2 3	YES
3 4	YES
1 2 2	YES
1 2 4 5	NO
	NO

### 3 Задача С. Быстрое прибавление [4 секунды, 256 mb]

Есть массив целых чисел длины  $n = 2^{24}$ , изначально заполненных нулями. Вам нужно сперва обработать  $m$  случайных запросов вида “прибавление на отрезке” по модулю  $2^{32}$ . Затем обработать  $q$  случайных запросов вида “сумма на отрезке” по модулю  $2^{32}$ .

#### Формат входных данных

На первой строке числа  $m, q$ . ( $1 \leq m, q \leq 2^{24}$ ). На второй строке пара целых чисел  $a, b$  от 1 до  $10^9$ , используемая в генераторе случайных чисел.

```
1. unsigned int cur = 0; // беззнаковое 32-битное число
2. unsigned int nextRand() {
3.     cur = cur * a + b; // вычисляется с переполнениями
4.     return cur » 8; // число от 0 до  $2^{24} - 1$ .
5. }
```

Каждый запрос первого вида генерируется следующим образом:

```
1. add = nextRand(); // число, которое нужно прибавить
2. l = nextRand();
3. r = nextRand();
4. if (l > r) swap(l, r); // получили отрезок [l..r]
```

Каждый запрос второго вида генерируется следующим образом:

```
1. l = nextRand();
2. r = nextRand();
3. if (l > r) swap(l, r); // получили отрезок [l..r]
```

Сперва генерируются запросы первого вида, затем второго.

#### Формат выходных данных

Выведите сумму ответов на все запросы по модулю  $2^{32}$ .

#### Примеры

fastadd.in	fastadd.out
5 5 13 239	811747796

#### 4    Задача D. Линейная сумма [3 секунды, 256 mb]

Есть  $n$  случайных точек на прямой с координатами от 0 до  $2^{32} - 1$ . У каждой точки есть значение от 0 до  $2^{32} - 1$ . Вам нужно обработать  $q$  случайных запросов вида “сумма значений точек, с координатами от  $l$  до  $r$  включительно”.

##### Формат входных данных

На первой строке числа  $n, q$ . ( $1 \leq n \leq 2^{20}, 1 \leq q \leq 2^{23}$ ). На второй строке пара целых чисел  $a, b$  от 1 до  $10^9$ , используемая в генераторе случайных чисел.

```
1. unsigned int cur = 0; // беззнаковое 32-битное число
2. unsigned int nextRand24() {
3.     cur = cur * a + b; // вычисляется с переполнениями
4.     return cur » 8; // число от 0 до  $2^{24} - 1$ .
5. }
6. unsigned int nextRand32() {
7.     unsigned int a = nextRand24(), b = nextRand24();
8.     return (a « 8) ^ b; // число от 0 до  $2^{32} - 1$ .
9. }
```

Каждая точка генерируется следующим образом:

```
1. value = nextRand32(); // значение точки
2. x = nextRand32(); // координата точки
```

Каждый запрос генерируется следующим образом:

```
1. l = nextRand32();
2. r = nextRand32();
3. if (l > r) swap(l, r); // получили отрезок [l..r]
```

Сперва генерируются точки, затем запросы.

##### Формат выходных данных

Выведите сумму ответов на все запросы по модулю  $2^{32}$ .

##### Примеры

linesum.in	linesum.out
5 5	3950632748
13 239	

##### Замечание

```
p = {value, x}
p[0] = {13, 41645}
p[1] = {7695587, 1253435649}
p[2] = {749170640, 2683600557}
p[3] = {2444595881, 1270561959}
p[4] = {3436107648, 486388002}
```

## 5 Задача Е. Длинное выражение [2 секунды, 256 mb]

Выведите значение заданного арифметического выражения.

### Формат входных данных

В первой строке входного файла задано выражение, состоящее из чисел, скобок и знаков бинарных операций. Каждое число в выражении это — целое неотрицательное число в промежутке от 0 до 10 000, включительно, записанное без ведущих нулей. Скобки бывают открывающие ('(') и закрывающие (')'). Операции задаются символами '+', '-', '\*' и '/'; знак умножения не может быть опущен. Гарантируется, что заданное выражение математически корректно, и результаты всех промежуточных операций — целые числа, не превышающие по модулю  $10^9$ . Выражение не содержит каких-либо других символов, в частности, пробелов. Длина выражения не меньше 1 и не больше 1 000 000 символов.

Учтите, что операции с одинаковым приоритетом при отсутствии скобок выполняются слева направо. Например, выражение  $a + b + c$  вычисляется как  $(a + b) + c$ .

### Формат выходных данных

В первой строке выходного файла выведите одно число — значение заданного выражения.

### Примеры

evalhard.in	evalhard.out
40-8/1*3	16
(5+50)/(2+3)	11

## Часть 2

### 6 Задача F. Точки и отрезки [0.5 секунд, 256 mb]

Дано  $n$  отрезков на числовой прямой и  $m$  точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка  $x$  считается принадлежащей отрезку с концами  $a$  и  $b$ , если выполняется двойное неравенство  $\min(a, b) \leq x \leq \max(a, b)$ .

#### Формат входных данных

Первая строка содержит два целых числа  $n$  ( $1 \leq n \leq 10^5$ ) — число отрезков и  $m$  ( $1 \leq m \leq 10^5$ ) — число точек. В следующих  $n$  строках записаны по два целых числа  $a_i$  и  $b_i$  — координаты концов соответствующего отрезка. В последней строке записаны  $m$  целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю  $10^9$ .

#### Формат выходных данных

В выходной файл выведите  $m$  чисел — для каждой точки выведите количество отрезков, в которых она содержится.

#### Примеры

segments.in	segments.out
2 2 0 5 7 10 1 6	1 0
1 3 -10 10 -100 100 0	0 0 1

## 7 Задача G. Обмен [0.5 секунд, 256 mb]

Пусть все натуральные числа исходно организованы в список в естественном порядке. Разрешается выполнить следующую операцию:  $swap(a, b)$ . Эта операция возвращает в качестве результата расстояние в текущем списке между числами  $a$  и  $b$  и меняет их местами.

Задана последовательность операций  $swap$ . Требуется вывести в выходной файл результат всех этих операций.

### Формат входных данных

Первая строка входного файла содержит число  $n$  ( $1 \leq n \leq 200\,000$ ) — количество операций. Каждая из следующих  $n$  строк содержит по два числа в диапазоне от 1 до  $10^9$  — аргументы операций  $swap$ .

### Формат выходных данных

Для каждой операции во входном файле выведите ее результат.

### Пример

swap.in	swap.out
4	3
1 4	1
1 3	4
4 5	2
1 4	