

## Содержание

Задачи	2
1 Задача А. Поиск пути на гриде	2
2 Задача В. Объединение отрезков	3
3 Задача С. Последовательность	4
Гроб ли это?	5
4 Задача D. Таблицы Юнга	5

## Задачи

### 1 Задача А. Поиск пути на гриде

Имя входного файла: `dfsongrid.in`  
Имя выходного файла: `dfsongrid.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Дано прямоугольное поле  $W \times H$ . Некоторые клетки проходимы, через некоторые ходить нельзя. Из клетки можно ходить в соседние по ребру (слева, справа, сверху, снизу).

Нужно из клетки  $(x_1, y_1)$  найти любой (не обязательно кратчайший, даже не обязательно простой) путь в клетку  $(x_2, y_2)$ .

#### Формат входных данных

На первой строке  $W, H, x_1, y_1, x_2, y_2$  ( $1 \leq x_1, x_2 \leq W \leq 1000, 1 \leq y_1, y_2 \leq H \leq 1000$ ). Далее  $H$  строк, в каждой из которых по  $W$  символов. Символ "." означает, что клетка проходима, а символ "\*" означает, что по ней ходить нельзя.

Клетки  $(x_1, y_1)$  и  $(x_2, y_2)$  не совпадают и обе проходимы.

#### Формат выходных данных

Если пути не существует, выведите NO.

Иначе выведите YES и последовательность клеток  $(x_i, y_i)$ , в которой первая совпадает с клеткой  $(x_1, y_1)$ , а последняя с клеткой  $(x_2, y_2)$ .

#### Пример

<code>dfsongrid.in</code>	<code>dfsongrid.out</code>
<code>4 2 1 1 4 2</code> <code>....</code> <code>....</code>	<code>YES</code> <code>1 1 2 1 3 1 4 1 3 1 3 2</code> <code>4 2</code>
<code>4 2 1 1 4 2</code> <code>..*</code> <code>.*</code>	<code>NO</code>
<code>4 2 1 1 4 2</code> <code>..*</code> <code>*...</code>	<code>YES</code> <code>1 1 2 1 2 2 3 2 4 2</code>

## 2 Задача В. Объединение отрезков

Имя входного файла: `merge.in`  
Имя выходного файла: `merge.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Решая задачу из контрольной по математике, Вася получил ответ в виде объединения  $N$  отрезков  $[L_i, R_i]$  на числовой прямой. Однако, некоторые из этих отрезков могут пересекаться друг с другом, что не слишком нравится Васе. Ваша задача — представить Васин ответ в виде объединения минимального количества отрезков.

### Формат входных данных

В первой строке указано число  $N$  ( $1 \leq N \leq 50000$ ). В следующих  $N$  строках перечислены пары целых чисел  $L_i$  и  $R_i$  ( $|L_i|, |R_i| \leq 50000$ ), каждая пара с новой строки, числа в парах отделены друг от друга одним или несколькими пробелами.

### Формат выходных данных

В первой строке выведите число  $M$  — количество отрезков в искомом объединении. В следующих  $M$  строках выведите сами эти отрезки в том же формате, что и во входном файле. Список отрезков необходимо упорядочить по возрастанию левого конца.

### Примеры

<code>merge.in</code>	<code>merge.out</code>
4	2
0 2	0 3
4 5	4 6
1 3	
5 6	

### 3 Задача С. Последовательность

Имя входного файла: `seq.in`  
Имя выходного файла: `seq.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 МБ

Дана последовательность целых чисел  $a_1, a_2, \dots, a_n$ , ( $1 \leq n \leq 100\,000$ ) каждое из которых по модулю не превосходит 10000. Эта последовательность записана на бумажной ленте, которая свернута в кольцо.

Разрежем это кольцо в некоторой точке между числами, получим полоску с записанной на ней последовательностью следующего вида:

$$a_j, a_{j+1}, \dots, a_n, a_1, a_2, \dots, a_{j-1}.$$

Назовем точку разреза хорошей, если все частичные суммы полученной последовательности строго положительны:

$$\begin{aligned} a_j &> 0 \\ a_j + a_{j+1} &> 0 \\ &\dots \\ a_j + a_{j+1} + \dots + a_n &> 0 \\ a_j + a_{j+1} + \dots + a_n + a_1 &> 0 \\ &\dots \\ a_j + a_{j+1} + \dots + a_n + a_1 + \dots + a_{j-2} &> 0 \\ a_j + a_{j+1} + \dots + a_n + a_1 + \dots + a_{j-2} + a_{j-1} &> 0 \end{aligned}$$

Вам требуется вычислить число хороших точек разреза.

#### Формат входных данных

Первая строка входного файла содержит число  $n$ , а во второй строке заданы числа  $a_1, a_2, \dots, a_n$ .

#### Формат выходных данных

Выведите в выходной файл число искомых хороших точек разреза.

#### Пример

<code>seq.in</code>	<code>seq.out</code>
5 0 1 -2 10 3	2

## Гроб ли это?

### 4 Задача D. Таблицы Юнга

Имя входного файла: `young.in`  
Имя выходного файла: `young.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Нужно найти число способов расставить числа от 1 до  $N$  внутри диаграммы Юнга площади  $N$  так, чтобы числа внутри каждой строки и каждого столбца возрастали. Каждое число от 1 до  $N$  нужно использовать ровно один раз.

$$1 \leq N \leq 50$$

#### Формат входных данных

Число строк диаграммы Юнга  $k$ . Далее  $k$  длин строк:  $a_1 \geq a_2 \geq \dots \geq a_k$ .

Число 1 должно оказаться в первой клетке первой строки.

Первый столбец диаграммы имеет высоту  $k$ . Высоты столбцов, также как и длины строк, убывают.

#### Формат выходных данных

Одно целое число — количество способов расставить числа от 1 до  $N$ , где  $N = \sum_{i=1}^k a_i$ .

#### Пример

<code>young.in</code>	<code>young.out</code>
2 2 2	2
4 4 3 2 1	768