

## Рампа

Андрея получила в подарок умные часы, в которых, помимо множества интересных функций, есть еще и альтиметр (функция измерения высоты). В поездке в горы Андрея решила протестировать эту функцию, измеряя высоту каждый километр. Прибыв в пункт назначения, Андрея проанализировала полученную последовательность высот длины  $N$ ,  $A = (A_1, A_2, \dots, A_N)$  и выделила некоторые подпоследовательности значений, которые она назвала *рампами*. *Рампа* состоит из чисел, не равных между собой, которые можно переставить таким образом, чтобы сформировать непрерывную последовательность последовательных чисел. Длина рампы – это количество элементов, из которых она состоит.

Например, для строки высот  $A = (3, 1, 6, 8, 4, 5, 6, 7, 4, 3, 4, 2)$  можно выделить рампы  $R1 = (8, 4, 5, 6, 7)$  длины 5 и  $R2 = (3, 4, 2)$  длины 3, так как при перестановке получаем  $R1 = (4, 5, 6, 7, 8)$  и  $R2 = (2, 3, 4)$ .

**Задание.** Разработайте программу, определяющую максимальную длину рампы и количество рамп максимальной длины.

**Входные данные.** Файл `rampa.in` содержит в первой строке целое число  $N$  – количество измерений, выполненных Андрея. Вторая строка файла содержит  $N$  целых чисел, разделённых пробелом, которые представляют значения высот, зарегистрированных Андрея.

**Выходные данные.** Файл `rampa.out` содержит 2 линии. В первой линии содержится максимальная длина рампы. Во второй линии содержится количество рамп максимальной длины.

**Ограничения.**  $1 \leq N \leq 5000$ ,  $1 \leq A_i \leq 1000$ ,  $1 \leq i \leq N$ . Время выполнения не превышает 0,1 секунды. Программа использует до 16 мегабайт оперативной памяти. Исходный файл решения называется `rampa.pas`, `rampa.c` или `rampa.cpp`.

### Пример.

`rampa.in`

11

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

`rampa.out`

5

2

**Объяснение:** В данном примере есть 2 рампы длины 5:  $R1 = (8, 4, 5, 6, 7)$  и  $R2 = (3, 4, 2, 1, 5)$