

## Minmax. Минимум/максимум

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

Реализуйте структуру данных для хранения целых чисел, поддерживающую следующие операции:

- добавить элемент;
- извлечь минимальный элемент;
- извлечь максимальный элемент.

(заметьте, что извлекаемый элемент удаляется)

### Формат входного файла

Первая строка входного файла содержит количество запросов  $n$  ( $1 \leq n \leq 100\,000$ ).  
Следующие  $n$  строк содержат запросы в следующем формате:

- `Insert(X)` — добавить число  $X$ ;
- `GetMin` — извлечь минимум;
- `GetMax` — извлечь максимум.

### Формат выходного файла

Для каждого запроса `GetMin` или `GetMax` требуется вывести соответствующий извлеченный элемент.

### Пример

minmax.in	minmax.out
11	3
Insert(3)	1
GetMin	-100
Insert(1)	8
GetMax	
Insert(-4)	
Insert(2)	
Insert(8)	
Insert(-100)	
GetMin	
GetMax	
Insert(2)	

## Keyins. Вставка ключей

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

Дан бесконечный массив, заполненный нулями. В него последовательно добавляются числа от 1 до  $n$ . Для каждого числа задан номер ячейки  $a_i$ . Если ячейка номер  $a_i$  содержит ноль, то число  $i$  записывается в нее. В противном случае ищется первая ячейка с большим номером, содержащая ноль, и число  $i$  записывается в нее.

Например, если мы хотим записать число  $i$  в третью ячейку, а ячейки со второй и четвертой заняты, а пятая ячейка свободна, то мы запишем  $i$  в пятую ячейку.

Требуется промоделировать указанный процесс.

### Формат входного файла

Первая строка входного файла содержит натуральное число  $n$  — количество вставок ( $n \leq 500\,000$ ). Вторая строка содержит  $n$  натуральных чисел  $a_1, \dots, a_n$  ( $a_i \leq 500\,000$ ).

### Формат выходного файла

Выведите в выходной файл единственное число — сумму произведений номеров ячеек на числа в них, взятую по модулю 12345.

### Пример

keyins.in	keyins.out
4	39
2 4 2 2	