

# Тренировки по алгоритмам 4.0 от Яндекса — Разминка

🕒 11 апр 2024, 19:28:27  
старт: 30 окт 2023, 12:00:00  
финиш: 27 ноя 2023, 00:59:59  
длительность: 27д. 12ч.  
...

Объявления жюри

📘 Ваше участие в соревновании завершено. Вы можете дорешивать задачи и отправлять решения вне соревнования

Положение участников Задачи Посылки

## А. Не минимум на отрезке

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Задана последовательность целых чисел  $a_1, a_2, \dots, a_n$ . Задаются запросы: сказать любой элемент последовательности на отрезке от  $L$  до  $R$  **включительно**, не равный минимуму на этом отрезке.

### Формат ввода

В первой строке содержатся два целых числа  $N, 1 \leq N \leq 100$  и  $M, 1 \leq M \leq 1000$  — длина последовательности и количество запросов соответственно.  
Во второй строке — сама последовательность,  $0 \leq a_i \leq 1000$ .

Начиная с третьей строки перечисляются  $M$  запросов, состоящих из границ отрезка  $L$  и  $R$ , где  $L, R$  - индексы массива, нумеруются с нуля.

### Формат вывода

На каждый запрос выведите в отдельной строке ответ — любой элемент на  $[L, R]$ , кроме минимального. В случае, если такого элемента нет, выведите "NOT FOUND".

#### Пример 1

Ввод	Вывод
10 5	NOT FOUND
1 1 1 2 2 2 3 3 3 10	2
0 1	NOT FOUND
0 3	10
3 4	3
7 9	
3 7	

#### Пример 2

Ввод	Вывод
4 2	NOT FOUND
1 1 1 2	2
0 2	
0 3	

Язык Python 3.9 (PyPy 7.3.11) ▾

Набрать здесь Отправить файл

```
1 class SegmentTree_Parameters:
2     def __init__(self, neutral, calc_op, change_op):
3         self.neutral = neutral
4         self.calc_op = calc_op
5         self.change_op = change_op
6
7 class SegmentTree():
8     def __init__(self, array: list, st_params: SegmentTree_Parameters) -> None:
9         def build_tree(array, index):
10             if 2*index+1 > len(array)-1: return
11
12             build_tree(array, 2*index+1)
13             build_tree(array, 2*index+2)
14             array[index] = (self.calc_op(array[2*index+1][0], array[2*index+2][0]), 0)
15
16         self.source_length = len(array)
17
18         self.neutral = st_params.neutral
19         self.calc_op = st_params.calc_op
20         self.change_op = st_params.change_op
21
22         if not array: raise Exception('Array is empty')
23
24         N = 1
25         while N < self.source_length:
26             N = N*2
27
28         self.tree = [(self.neutral, self.neutral)]*(2*N-1)
29
30         for i in range(self.source_length):
31             self.tree[N-1+i] = (array[i], self.neutral)
32
33         build_tree(self.tree, 0)
34
35     def getSegment(self, left, right):
36     def get(l, r, i, li, ri):
37         if l > ri or r < li: return self.neutral
38
```

Отправить 📘 осталось 100 попыток

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
17 янв 2024, 20:03:18	105136971	A	Python 3.9 (PyPy 7.3.11)	OK	-	0.545s	31.74Mb	-	-	<a href="#">отчёт</a>
31 окт 2023, 17:10:37	95382911	A	Python 3.9 (PyPy 7.3.11)	OK	-	264ms	28.33Mb	-	-	<a href="#">отчёт</a>
31 окт 2023, 16:58:46	95380850	A	Python 3.9 (PyPy 7.3.11)	WA	-	209ms	28.09Mb	5	-	<a href="#">отчёт</a>