

ВШЭ АиСД 2021. Стек, очередь (noSTL)

4 дек 2021, 18:59:01
старт: 30 сен 2021, 11:30:00
финиш: 10 окт 2021, 23:59:59
длительность: 10д. 12ч.
начало: 30 сен 2021, 11:30:00
конец: 10 окт 2021, 23:59:59

С. Пробки в городе (0.3)

Ограничение времени	2 секунды
Ограничение памяти	64.0 Мб
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

В городе N очень часто образуются пробки. Так как жителей в городе много, пробки иногда получаются довольно длинными. Мэрия города N решила установить некоторые правила, чтобы её сотрудники могли быстрее добираться до места работы.

Обычные автомобилисты встают в конец пробки. Однако сотрудники мэрии имеют право вставать ровно в ее середину, причем при нечетной длине пробки они встают сразу за центром.

Так как мэрия не доверяет жителям города, вас попросили разработать программу, которая бы отслеживала порядок автомобилистов в пробке.

Формат ввода

В первой строке входных данных записано число N ($1 \leq N \leq 10^5$) - количество запросов к программе. Следующие N строк содержат описание запросов в формате:

- " $+$ i " - житель с номером i ($1 \leq i \leq N$) встает в конец пробки.
- " $*$ i " - сотрудник мэрии с номером i встает в середину пробки.
- " $-$ " - первый житель из пробки проезжает пробку. Гарантируется, что на момент такого запроса пробка не пуста.

Формат вывода

Для каждого запроса типа " $-$ " программа должна вывести номер жителя, который выезжает из начала пробки. Порядок вывода соответствует изначальному порядку следования запросов во входных данных.

Пример 1

Ввод	<input type="text"/>	Вывод	<input type="text"/>
7		1	
+ 1		2	
+ 2		3	
-			
+ 3			
+ 4			
-			
-			

Пример 2

Ввод	<input type="text"/>	Вывод	<input type="text"/>
------	----------------------	-------	----------------------

Ввод Вывод

10

+ 1

+ 2

* 3

-

+ 4

* 5

-

-

-

-

1

3

2

5

4

Язык (make) GCC C++17

Набрать здесь

Отправить файл

```
1 #include <iostream>
2 #include <string>
3
4 class Dequeue {
5 private:
6     int64_t* dequeue;
7
8     int indexStart;
9     int indexEnd;
10
11     int cursor;
12
13 public:
14     explicit Dequeue(int64_t dequeueSize) {
15         dequeue = new int64_t[dequeueSize];
16
17         cursor = -1;
18     }
19
20     void Pop_front() {
21         std::cout << dequeue[cursor] << std::endl;
22
23         dequeue[cursor] = NULL;
24
25         --cursor;
26     }
27
28     void Push_centre(int n) {
29         int centre = ((cursor + 1) / 2);
30
31         for (int i = ++cursor; i > centre; --i) {
32             dequeue[i] = dequeue[i - 1];
33         }
34
35         dequeue[centre] = n;
36     }
37
38     void Push_back(int n) {
```

Отправить

Предыдущая

Следующая