Java → Cycled list

Hard 9 minutes

Code Challenge — Write a program

The aim of this task is to slightly modify a doubly linked list and make a **circular doubly linked list**. It means that the previous element for the head is the tail and the next element for the tail is the head.

You are given a circular list. Initially, the main pointer is on the first element. You receive commands for removing elements at different distances to the left or the right from the current one. Next commands count from the place where an element was removed.

Input: In the first line divided by spaces are two numbers $N < 1000$ and $K < 1000$ – the number of elements in the list and the number of commands. In the second line divided by spaces is the list. In the following K lines are commands of two types:

- «r value» removes the element, which is on the right by 'value' elements;
- «l value» removes the element, which is on the left by 'value' elements.

Output: In the first line, print the resulting list after all removals. The first element must be to the left of all other elements of the resulting list in the initial list.

Sample Input 1:

```
5 2
1 2 3 4 5
r 1
r 1
```

Sample Output 1:

```
1 3 5
```

Code Editor IDE

Solve in IDE

IDE is opened If you don't see your IDE opened, switch to it manually

Code snippets from theory

Postpone problem

Time limit: 8 seconds Memory limit: 256 MB

Comments (1) Hints (0) Useful links (0) Solutions (0)

SK Share something, Sergey Kubatko

Post Please do not post solutions here

C **cypherman** about 1 month ago Report

Once again lacking clarification on how you want pointers to work. Poor description of tasks including an absence of examples leads to 0 people solving this problem.

Code snippets from theory

1 Reply

Code snippets from theory