CS1020E | Lab 4 | Exercise 2

Keyboard

Objective

The objective of this exercise is to learn how to implement and use a **Doubly Linked List**.

Problem Description

You are implementing a text editor. Since a text editor can be a complex piece of software, you are going to simplify things. The first simplification is the user's text will only consist of one line. Second, the user will not do any operation other than:

1. Press the left button *K* times. The cursor will move to the left *K* times. While moving, if the cursor reaches the beginning of the line, it will not move further.
2. Press the right button *K* times. The cursor will move to the right *K* times. While moving, if the cursor reaches the end of the line, it will not move further. Note that the end of the line is one position to the right of the last character of the line.
3. Insert a character *X*. A character *X* (note that *X* is a variable) will be inserted at the current cursor position. After this operation, the cursor will be located one position to the right of the newly inserted character.
4. Delete a character. A character that is located on the left of the cursor will be deleted. If the cursor is located in the beginning of the line, then the operation will have no effect. **After this operation, the cursor will be located at the beginning of the line**.

After all operations are executed, you must output the line of text to the user.

Add your code only to the parts of the files indicated. Do not modify any other part of the given code, and do not add new files.

Inputs

The first line of the input consists of a single integer *N*, the number of user's operations.

N lines follow. Each of the lines specifies an operation, in one of the following forms:

- `left` *K* — specifying Operation 1
- `right` *K* — specifying Operation 2
- `insert` *X* — specifying Operation 3
- `delete` — specifying Operation 4

The values of *N* and *K* are between 1 and 100 inclusive, and *X* is a single lowercase letter.

Outputs

The final text line after all of the user's operations.

Sample Input

```
7
insert o
insert n
left 10
insert j
right 1
delete
insert a
```

Sample Output

```
ajn
```

Explanation

Below is the result of the text after each operation. The underlined and bold character indicates the position of the cursor.

o_

on_

on

j**o**n

jo**n**

jn

a**j**n

Hint

Consider adding a dummy head node at the beginning of the line to make insertion and deletion of node easier. A dummy tail node can also be added to represent the end of the line.

Submission

You need to submit **ALL** your completed skeleton **`*.cpp`** and **`*.h`** files to CodeCrunch (https://codecrunch.comp.nus.edu.sg/) before the specified deadline. We will take only your latest submission.

Late submissions will not be accepted. The submission system in CodeCrunch will automatically close at the deadline.