Text Editor

locked

This is a take-home challenge.

In this challenge, you need to implement a simple text editor. The editor stores a string of characters, and also highlights a special position in the string called the cursor. The editor should support the following operations, and redisplay the current text (along with the cursor) after performing any of them:

- **left**: move cursor left by one character (or do nothing if at the beginning)
- **right**: move cursor right by one character (or do nothing if at the end)
- **delete**: delete the character at the cursor (or do nothing if there is no character at the cursor). After deletion, the cursor will move to the next character on the right of the deleted character. If there is no such character, the cursor will be move to the left of the deleted character.
- **insert *c***: insert the character ***c*** just after the cursor, and move the cursor to the newly inserted character.

Approach: You must use a doubly linked list for storing the string. Different editor operations can then be implemented as insert/delete/move operations in the linked list.

Input Format

- The first line will contain N, the number of commands to the editor, following which there will be N lines, each containing an editor command using the following format.
- 1 indicates `left` operation.
- 2 indicates `right` operation.
- 3 indicates `delete` operation.
- 4 c indicates `insert c` operation

Constraints

$1 \leq N \leq 10^6$

Output Format

- For each command, you need to print the state of the editor.
- The state of the editor should be displayed using 2 lines, where the first line contains the string, and the second line indicates the cursor position, by placing the letter C. For example, if the current string is 'Text editor' and the cursor is on the character 'x', you need to print the following:

```
Text Editor
  C
```

- Initially, the string will be empty, and the cursor will be at the first position. You should also print the initial state.

- Hence, there should be a total of 2N + 2 lines in the output.

Sample Input 0

```
6
1
4 T
4 e
4 t
1
4 x
```

Sample Output 0

```
C

C
T
C
Te
 C
Tet
  C
Tet
 C
Text
  C
```

Sample Input 1

```
4
4 a
1
1
4 b
```

Sample Output 1

```
C
a
C
a
C
a
C
ab
 C
```

Sample Input 2

```
9
4 h
4 e
4 l
4 l
4 o
2
1
2
2
```

Sample Output 2

```
C
h
C
he
 C
hel
  C
hell
   C
hello
    C
hello
    C
hello
   C
hello
    C
hello
    C
```

Sample Input 3

```
8
4 h
4 e
4 l
1
3
3
2
4 i
```

Sample Output 3

```
C
h
C
he
 C
hel
  C
hel
 C
hl
 C
h
C
h
C
hi
 C
```

Sample Input 4

```
11
4 h
4 e
4 l
3
3
2
4 i
1
3
3
4 a
```

Sample Output 4

```
C
h
C
he
 C
hel
  C
he
 C
h
C
h
C
hi
 C
hi
C
i
C

C
a
C
```

Sample Input 5

```
7
1
2
3
4 H
4 H
1
3
```

Sample Output 5

```
C

C

C

C
H
C
HH
 C
HH
C
H
C
```

Sample Input 6

```
25
1
2
3
4 m
1
2
4 a
1
2
4 d
```

```
1
2
4 a
1
2
4 m
1
1
3
1
3
3
1
3
3
```

Sample Output 6

```
C

C

C

C
m
C
m
C
m
C
ma
 C
ma
C
ma
 C
mad
  C
mad
 C
mad
  C
mada
   C
mada
  C
mada
   C
madam
    C
madam
   C
madam
  C
maam
  C
maam
 C
mam
 C
mm
 C
mm
C
m
C

C
```

Sample Input 7

```
18
4 i
4 m
4 a
4 d
1
1
1
4 i
4 i
3
1
4 t
2
2
2
4 r
4 a
4 s
```

Sample Output 7

```
C
i
C
im
 C
ima
  C
imad
   C
imad
  C
imad
 C
imad
C
iimad
 C
iiimad
  C
iimad
  C
iimad
 C
iitmad
  C
iitmad
   C
iitmad
    C
iitmad
     C
iitmadr
      C
iitmadra
       C
iitmadras
        C
```

Submissions: 87
Max Score: 25
Difficulty: Medium

Rate This Challenge:

More

C++

```cpp
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;
struct letter
{
    char ch;
    struct letter * next;
    struct letter * prev;
};
void left(struct letter** pntr, struct letter* head, struct letter* tail,int* s, int* count)
{
    if ((*count)!=0)
    {
        if ((*s)!=0)
        {
            *pntr=(*pntr)->prev;
            (*s)--;
        }
    }
}
void right(struct letter** pntr, struct letter* head, struct letter* tail,int* s, int* count)
{
    if ((*pntr)->next!=tail)
    {
        *pntr=(*pntr)->next;
        (*s)++;
    }
}
void del(struct letter** pntr, struct letter* head, struct letter* tail,int* s, int* count)
{
    if ((*pntr)!=head)
    {
        (*pntr)->prev->next=(*pntr)->next;
        (*pntr)->next->prev=(*pntr)->prev;
        if ((*pntr)->next!=tail)
        {
            *pntr=(*pntr)->next;
        }

        else
        {
            *pntr=(*pntr)->prev;
            (*s)--;
        }
        (*count)--;
    }
}
void insert(struct letter** pntr, char c, struct letter* head, struct letter* tail,int* s, int*
count)
{
    if ((*pntr)!=head)
    {
        struct letter* pt = new struct letter;
        pt->ch=c;
        (*pntr)->next->prev=pt;
        pt->next=(*pntr)->next;
        pt->prev=(*pntr);
        (*pntr)->next=pt;
        *pntr=pt;
        (*s)++;
```

```
    }
    else
    {
        struct letter* pt = new struct letter;
        pt->ch=c;
        (*pntr)->next=pt;
        pt->prev=*pntr;
        pt->next=tail;
        tail->prev=pt;
        *pntr=pt;
    }
    (*count)++;
}
void printl(struct letter* head,int s, int count)
{
    struct letter* x = head->next;
    for (int hh=1; hh<=count; hh++)
    {
        cout<<x->ch;
        x=x->next;
    }
    cout<<endl;
    for (int hh=1; hh<s+1; hh++)
    {
        cout<<" ";
    }
    cout<<"C";
    cout<<endl;
}


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    int s=0;
    int count=0;
    int n,m;
    cin>>n;
    char f;
    struct letter* head=new struct letter;
    struct letter* ptr = head;
    struct letter* tail= new struct letter;
    head->next=tail;
    tail->prev=head;
    cout<<endl;
    cout<<"C"<<endl;
    while(n--)
    {
        cin>>m;
        if (m==1)
        {
            left(&ptr,head,tail,&s,&count);
            printl(head,s,count);
        }
        else if (m==2)
        {
            right(&ptr,head,tail,&s,&count);
            printl(head,s,count);
        }
        else if (m==3)
        {
            del(&ptr,head,tail,&s,&count);
            printl(head,s,count);
        }
        else if (m==4)
        {
            cin>>f;
```

```
            insert(&ptr, f,head,tail,&s,&count);
            printl(head,s,count);
        }


    }
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code