

SBE201 Linked List

In this problem you will be implementing a linked list of integers (int data type) in strict C language. The main implemented operations are insertion, deletion, and printing of the list.

Input

The input consists of lines starting with a single character. The character at the beginning of each line represents the operation to be done. Below is the description of the operations that should be implemented.

- **f N** : this operation is insertion at the front (head) of the linked list. The integer N is the value to be inserted. Example: f 7 add 7 to the front.
- **i $M N$** : this operation is insertion at index M . The index is zero-based. The value to be inserted is the integer N . if the index is beyond the end of the list, the value should be inserted as the last element. Negative indexes are not expected in the input so you do not have to check for it. Example: i 5 8 insert 8 at index 5 such that the element at 5 is 8 after successful insertion.
- **r**: this operation is deletion at the front. If the list is empty, nothing should happen to the list.
- **d M** : this operation is deletion at an index M . The index is zero-based. If the specified index is beyond the end of the list, nothing should happen to the list. Negative indexes are not expected in the input so you do not have to check for it. Example: d 5 means to delete the element at index 5.
- **q**: this operation is to stop the program and exit

HINT: do not forget to perform `scanf("%c",&tmp)` at the end of each input line to read the '\n' character and remove it from the input buffer.

Output

The output of the program is one line for each line of the input. This line should print the complete linked list, node by node, separated by single spaces. For example, the linked list $5 \rightarrow 7 \rightarrow 2 \rightarrow 8 \rightarrow \text{NULL}$ should be printed as 5 7 2 8. If the list is empty, the program should print the string "empty" all in lower case with no spaces in one line.

Example

Input:

```
r
f 3
f 9
i 1 5
i 9 7
r
d 1
d 0
r
f 13
r
r
r
```

r
q

Output:

empty
3
9 3
9 5 3
9 5 3 7
5 3 7
5 7
7
empty
13
empty
empty
empty
empty