

## CSC240 – C Calculator with Stacks/Linked Lists

Using C, write a program that will act as a simple interactive calculator. Your calculator should include +, -, and \*. Your calculator should also have "undo" and "redo" functionality.

Implement the "undo" and "redo" functionality using two Stacks implemented with linked lists. Each node in both stacks will include an operator and integer value. Push and pop operations should not be duplicated for the two stacks.

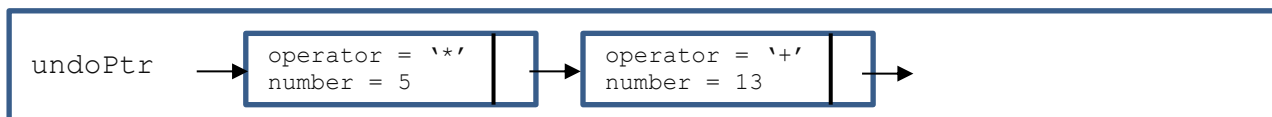
### Sample Output:

```
C Calculator
+-----+
Enter starting value: 7
Enter next operation (u-Undo, r-Redo, q-Quit): + 13

Result = 20

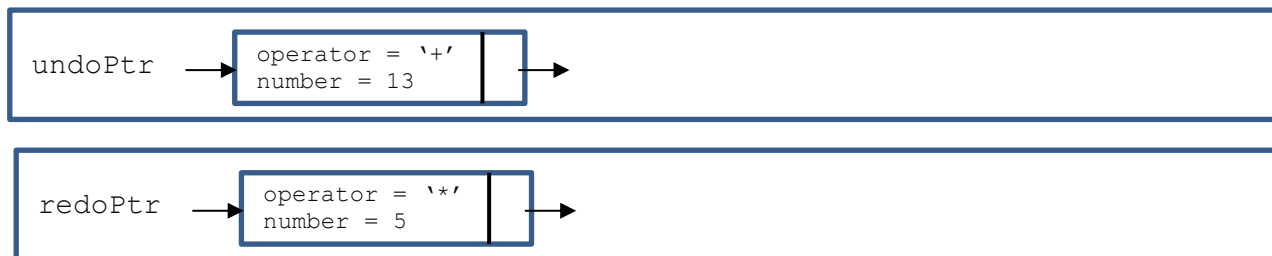
Enter next operation (u-Undo, r-Redo, q-Quit): * 5

Result = 100
```



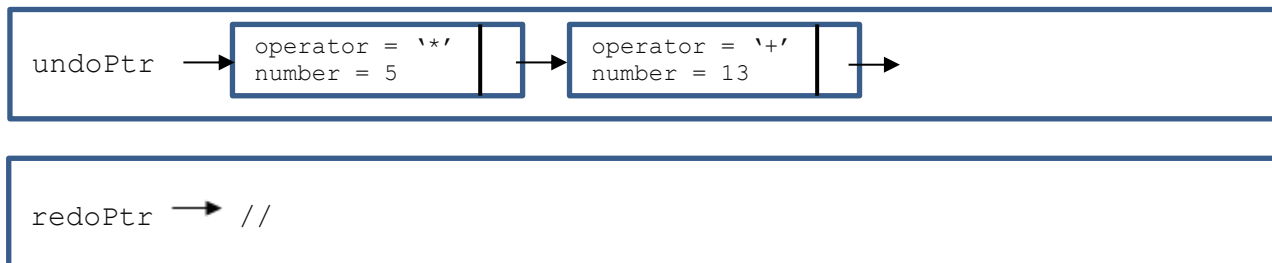
```
Enter next operation (u-Undo, r-Redo, q-Quit): u

Result = 20
```



```
Enter next operation (u-Undo, r-Redo, q-Quit): r

Result = 100
```



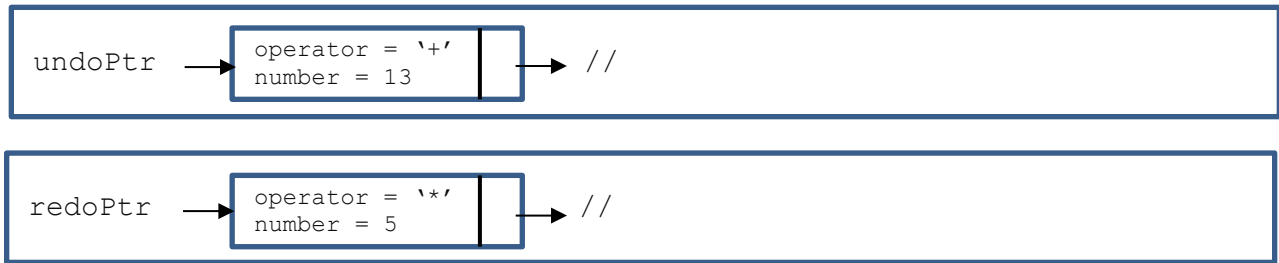
```
Enter next operation (u-Undo, r-Redo, q-Quit): / 20
***Not a valid operation.

Enter next operation (u-Undo, r-Redo, q-Quit): y
***Not a valid operation

Enter next operation (u-Undo, r-Redo, q-Quit): r
***Not a valid operation
```

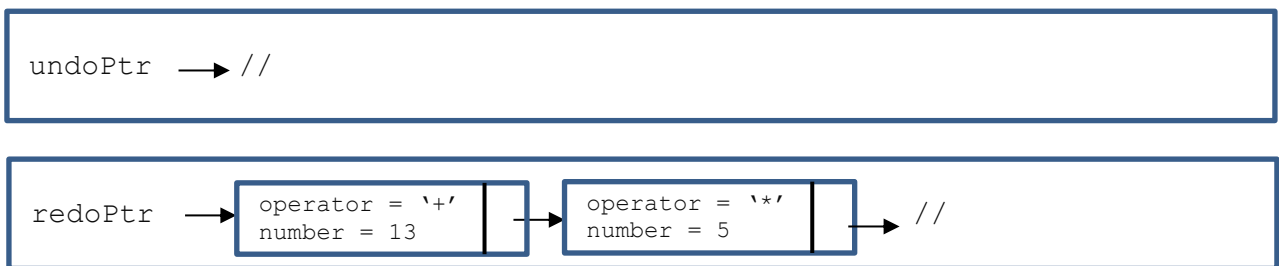
Enter next operation (u-Undo, r-Redo, q-Quit): u

Result = 20



Enter next operation (u-Undo, r-Redo, q-Quit): u

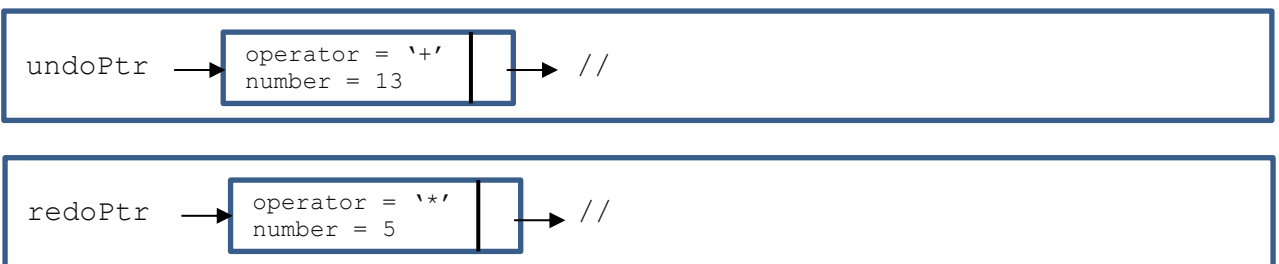
Result = 7



Enter next operation (u-Undo, r-Redo, q-Quit): u  
\*\*\*Not a valid operation

Enter next operation (u-Undo, r-Redo, q-Quit): r

Result = 20



Enter next operation (u-Undo, r-Redo, q-Quit): q

### **Program design:**

The main() function should be an outline of what the program does. All “details” of the tasks should be placed in functions.

- first make a list of tasks the program will need to perform
- decide which tasks to make into functions
- draw a structure chart to outline your program

Submit your program **AND** structure chart through Lab4 Dropbox in Canvas.