Inverting Stack

Modify your stack implementation as per the following specifications

push(value): Reverse the order of existing elements in the stack and then push new element onto it

pop(): Pop the top element from the stack and then reverse the order of remaining elements in the stack

Rest all specifications about the stack remain as we have discussed before in the class.

Write your own function that reverses the order of elements in the stack.

Similarly, write your own function that prints the elements in the stack.

Codes for various commands

1: Create new stack

2: Push

3: Pop

7: Print Stack contents

Explanation of the example input and output file

Line 1 of the input file has code 1. It means that we have to create a new stack. The capacity for the new stack is specified on the next line. The capacity is 5 elements.

Line 3 of the input file has code 2. It means that we have to push a value to the stack. The value to be pushed is provided on the next line. We have to push 50 to the stack.

Line 5 of the input file has code 2. It means that we have to push a value to the stack. The value to be pushed is provided on the next line. We have to push 71 to the stack.

Line 7 of the input file has code 7. It means that we have to print the status of the stack. Corresponding status is written as line 1 in output file.

Line 8 of the input file asks us to push 99 to the stack.

Line 10 asks us to print the status of the stack. Corresponding status is the line 2 of the output file.

Note that order of existing elements was reversed before pushing the new element.

Line 11 asks us to push 125 to the stack.

Line13 asks us to push 192 to the stack.

Line 15 asks us to print the status of the stack. Corresponding status is the line 3 of the output file.

Note that the order of elements was reversed twice: once before pushing 125 and once before pushing 192.

Line 16 asks us to pop a value from the stack. The value 192 will be removed from the stack and the order of remaining elements will be reversed.

Line 17 asks us to print the status. It is the line 4 of the output file.

Similarly, after processing lines 18 and 19 of the input file, value 125 will be removed from the stack and the status will be written as line 5 of the output file.