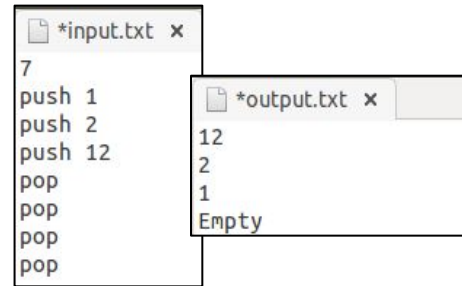# Lab 3 Stack, Queue

2019. 03. 21

lab 3-1. Stack

- You can implement Stack in any way you want (array or linked list).
- You will use only Push and Pop in this Lab (other functions may be needed for Push and Pop).
  - maximum size of stack is 100 (in case you use array).
  - push x : push element x to stack (Note that x is int value).
    - If stack is full, then print "Full" (in case you use array).
  - pop : pop an element from stack and print the element.
    - If stack is empty, then print "Empty".
- You have to use file I/O like the previous assignment.

```
 *input.txt  ×
7
push 1
push 2
push 12
pop
pop
pop
pop
```

```
 *output.txt  ×
12
2
1
Empty
```

lab 3-1. Stack

- Submission
  - Project directory name : lab3-1
  - Source file name : p3-1.c
  - Executable file name : p3-1.out
  - You should upload in the hconnect (Gitlab) server.
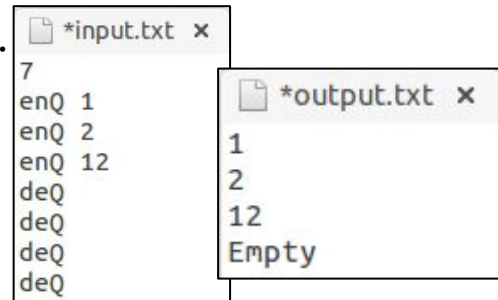 - If you are not familiar with using git, please read the manual. ([GitLab 사용법.pdf](#))

```
daewook@daewook-VirtualBox:~/2019_CSE2010_123456789/lab3-1$ gcc p3-1.c -o p3-1.out
daewook@daewook-VirtualBox:~/2019_CSE2010_123456789/lab3-1$ ./p3-1.out
daewook@daewook-VirtualBox:~/2019_CSE2010_123456789/lab3-1$ cat output.txt
12
2
1
Empty
```

Lab 3-2 Queue

- You can implement Queue in array.
- You will use only Enqueue and Dequeue in this Lab (other functions may be needed for Enqueue and Dequeue).
  - maximum size of queue is 100.
  - enQ x : Enqueue element x to queue (Note that x is int value).
    - If queue is full, then print "Full".
  - deQ : Dequeue an element from queue and print the element.
    - If queue is empty, then print "Empty".
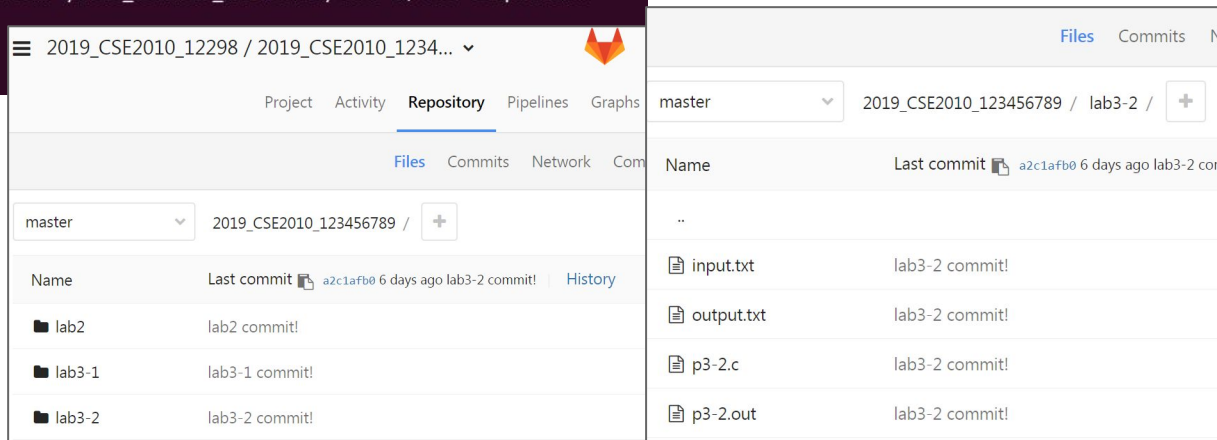- You have to use file I/O like the previous assignment.

*input.txt

```
7
enQ 1
enQ 2
enQ 12
deQ
deQ
deQ
deQ
```

*output.txt

```
1
2
12
Empty
```

Lab 3-2 Queue

- Submission
  - Project directory name : lab3-2
  - Source file name : p3-2.c
  - Executable file name : p3-2.out
  - You should upload in the hconnect (Gitlab) server.

Lab 3-3 Infix to Postfix conversion & evaluation

- Obtain an infix expression from the given input file. The expression ends with #.
  - Available operators: +, -, *, /, %, (,)
  - Operands: single-digit numbers (1,2,3,4,5,6,7,8,9).
  - Conditions:
    - The expression should be no more than 100 characters.
    - You have to use Stack Data Structure you implemented in Lab 3-1 (Note that you may change Element type from int to char).
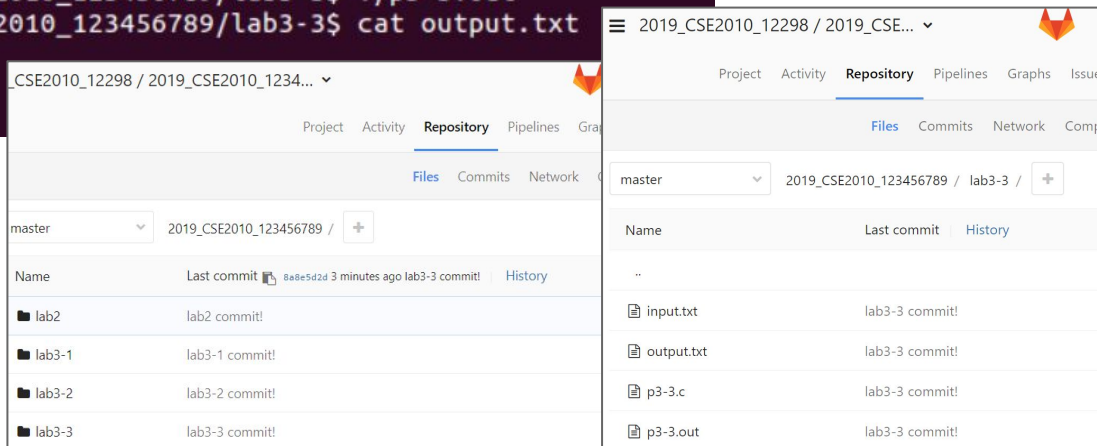- Operator precedence:

Token	precedence
* / %	2
+ -	1

- You have to use file I/O like the previous assignment.

```
*input.txt ×
4*(7+3%6)-(4/2)+9-(2*3)#
```

```
*output.txt ×
Infix Form : 4*(7+3%6)-(4/2)+9-(2*3)
Postfix Form : 4736%+*42/-9+23*-
Evaluation Result : 41
```

Lab 3-3 Infix to Postfix conversion & evaluation

- ● Submission
  - ○ Project directory name : lab3-3
  - ○ Source file name : p3-3.c
  - ○ Executable file name : p3-3.out
  - ○ You should upload in the hconnect (Gitlab) server.

DeadLine

Sunday, 31 March, 23 : 59 pm