



카테고리 없음

[알고리즘](Python) 백준 10828 - 스택

그리버 | 2023. 9. 12. 15:15 | 수정 | 삭제

유익사항	북유럽 4국(덴마크/노르웨이/스웨덴/핀란드) 8~11일 #베스트셀러	서유럽 3~4국 8일/9일/10일 #실
4,890,000원~	2,599,000원~	
3,090,000원~	발칸 2국 (크로아티아/슬로베니아) 9일 #두브로브니크 #플리트비체 #2국 깊이보기	[연합상품]양구 1편치볼둘레길트근미술관 #상무#습밥중식포함
9,990,000원~	내생에 꼭 한번! 아프리카 완전일주 #리얼사파리 #세계3대폭포 #희망봉	[노팁/노쇼핑 -스테디항공일주 10일 #빅토#퀘벡 숙박 #밴프 숙박포부 #와이너리 투
	35,900원~	
	6,499,300원~	

<https://www.acmicpc.net/problem/10828>

EKJOON ONLINE JUDGE	10828번: 스택 www.acmicpc.net
------------------------	-------------------------------

문제

주제	스택
시간 / 메모리 제한	0.5초 / 256 MB
정답 비율	37.231%

정수를 저장하는 스택을 구현한 다음, 입력으로 주어지는 명령을 처리하는 프로그램을 작성하시오.
명령은 총 다섯 가지이다.

- push X:
 - 정수 X를 스택에 넣는 연산
- pop:
 - 스택에서 가장 위에 있는 정수를 빼고, 그 수를 출력한다.
 - 만약 스택에 들어있는 정수가 없는 경우에는 -1을 출력한다.
- size:

- 스택에 들어있는 정수의 개수를 출력한다.
- empty:
 - 스택이 비어있으면 1, 아니면 0을 출력한다.
- top:
 - 스택의 가장 위에 있는 정수를 출력한다.
 - 만약 스택에 들어있는 정수가 없는 경우에는 -1을 출력한다.

입력

- 첫째 줄에 주어지는 명령의 수 N ($1 \leq N \leq 10,000$)이 주어진다.
- 둘째 줄부터 N 개의 줄에는 명령이 하나씩 주어진다.
 - 주어지는 정수는 1보다 크거나 같고, 100,000보다 작거나 같다.
 - 문제에 나와있지 않은 명령이 주어지는 경우는 없다.

출력

- 출력해야 하는 명령이 주어질 때마다, 한 줄에 하나씩 출력한다.

- 예제 입력 1

```
14
push 1
push 2
top
size
empty
pop
pop
pop
size
empty
pop
push 3
empty
top
```

- 예제 출력 1

```
2
2
0
2
1
-1
0
1
-1
0
3
```

- 예제 입력 2

```
7
pop
top
push 123
top
pop
top
pop
```

- 예제 출력 2

```
-1
-1
```

```
123
123
-1
-1
```

스택 개념

스택(stack)은 맨 마지막에 들어온 데이터가 맨 처음 나가는 LIFO(Last In First Out) 구조이다.
스택의 ADT는 다음과 같다.

```
### 스택의 ADT ###
Stack
# Object
    : LIFO 객체
# Operation
    1. empty() -> Boolean
    : 스택이 비어 있으면 True, 아니면 False 반환
    2. push(data)
    : data를 스택의 맨 위에 삽입
    3. pop() -> element
    : 스택의 맨 위에 있는 데이터를 삭제하며 반환
    4. peek() -> element
    : 스택의 맨 위에 있는 데이터를 반환만 함
```

파이썬에서 큐(queue)는 큐 모듈을 따로 지원하는 반면, 스택은 지원하지 않는다. 그 이유는 파이썬 기본 자료형인 리스트를 이용해서도 간단히 구현할 수 있기 때문이다.

리스트의 append()를 이용해 큐 ADT의 push()를 구현할 수 있다.

그리고 리스트의 pop()을 이용해 큐 ADT의 pop()을 구현할 수 있다.

정답 풀이

```
class Stack:

    def __init__(self):
        self.container = list()
        self.size = 0

    def empty(self):
        if not self.container:
            return 1
        else:
            return 0

    def push(self, data):
        self.container.append(data)
        self.size += 1

    def pop(self):
        if self.empty():
            return -1
        self.size -= 1
        return self.container.pop()

    def top(self):
        if self.empty():
            return -1
        return self.container[-1]

import sys

N = int(sys.stdin.readline())

queue = Stack()

for i in range(N):
    line = sys.stdin.readline().split()
```

```
if line[0] == 'push':
    queue.push(line[1])

elif line[0] == 'pop':
    print(queue.pop())

elif line[0] == 'size':
    print(queue.size)

elif line[0] == 'empty':
    print(queue.empty())

elif line[0] == 'top':
    print(queue.top())
```

공감

☐ Secret

안녕하세요! 어떤 댓글이든 환영합니다! 🍷

댓글달기