

# [읽을거리] 5. 스택/큐

## 시간복잡도

Queue의 종류별 시간복잡도는 다음과 같습니다.

	offer()	peak()	poll()	size()
PriorityQueue	$O(\log n)$	$O(1)$	$O(\log n)$	$O(1)$
LinkedList	$O(1)$	$O(1)$	$O(1)$	$O(1)$
ArrayDeque	$O(1)$	$O(1)$	$O(1)$	$O(1)$
ConcurrentLinkedQueue	$O(1)$	$O(1)$	$O(1)$	$O(n)$
ArrayBlockingQueue	$O(1)$	$O(1)$	$O(1)$	$O(1)$
PriorityBlockingQueue	$O(\log n)$	$O(1)$	$O(\log n)$	$O(1)$
SynchronousQueue	$O(1)$	$O(1)$	$O(1)$	$O(1)$
DelayQueue	$O(\log n)$	$O(1)$	$O(\log n)$	$O(1)$
LinkedBlockingQueue	$O(1)$	$O(1)$	$O(1)$	$O(1)$

Stack은 class 로 제공되고 있으므로 종류가 없고 시간복잡도는 모두  $O(1)$ 을 갖습니다.

## Deque 는 어떻게 읽을까?

### Pronunciation [ edit ]

- IPA<sup>(key)</sup>: /dɛk/
- Rhymes: -ɛk
- Homophone: deck

### Noun [ edit ]

deque (plural **deques**)

1. (*computing*) A linear [data structure](#) in which elements may be appended to or removed from either end. [quotations ▼]

Coordinate terms: [list](#), [stack](#), [queue](#)

*This algorithm is difficult to implement with a standard queue, but with a **deque** it's easy.*

<https://en.wiktionary.org/wiki/deque>

**Deque** 는 `double-ended queue` 의 약자로 발음기호대로 읽으면 [덱] 으로 발음됩니다.

[덱]으로 읽는 것이 정상적이 발음이며, 스펠링의 생김때문에 [디-큐]라고 읽는 경우도 있습니다.

java에서 큐에서 값을 빼는 것은 보통 poll 이라고 부르고 있지만, computer science 에서 큐에서 값을 빼는 행위를 dequeue 라고 합니다. [디-큐]로 읽으면 dequeue와 발음이 같아져서 혼란을 줄 수 있으므로 [덱]으로 발음하는 것이 좋습니다.

👤 그럼에도 개발자들 사이에서는 [디-큐], [데크], [덱] 등으로 읽는 방법이 혼재되어있는 현실입니다.

## Stack 은 어디에 사용될까?

### Function Call

다음과 같은 코드가 있다고 합시다.

```
void func1() {
    func2();
}

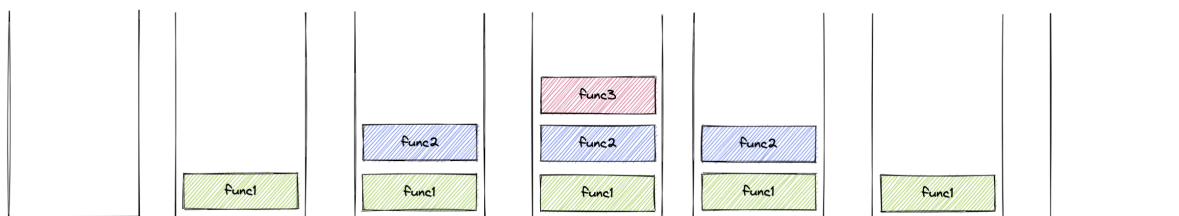
void func2() {
    func3();
}

void func3() {
    // do something
}
```

func1 을 실행하면 func2가 실행되고, func2에 의해서 func3이 실행됩니다.

func3이 종료되면, func2로 돌아가 수행을 마무리 하고, func1로 돌아갑니다.

그림으로 그려보면 이렇게 동작하게 되죠.



함수가 호출될 때 현재의 실행상태(context)를 stack에 저장하고 함수를 수행하게 됩니다. 함수의 실행이 종료되면 stack에서 pop 하게 되죠.

만약 함수가 종료되지 않은채로 다음 함수를 호출하는 과정이 무한으로 반복하게 되면, 언젠가 Stack이 저장할 수 있는 한계를 넘어서게 되겠죠.

이 경우 발행하는 에러를 **Stack Overflow** 라고 합니다.

## Do/Undo

Ctrl+Z (혹은 Cmd+Z) 를 눌러서 되돌리기(Undo) 하는 것은 아주 익숙한 기능이죠. 이 Undo 기능을 개발할 때도 Stack을 활용합니다.

컴퓨터로 수행하는 작업들을 하나 하나씩 Stack에 저장했다가 되돌리기 할 때마다 Stack에서 pop 해서 방금 전에 수행했던 명령을 취소하는 방식이죠.

## Screen Stack

휴대폰 앱을 사용하는 것은 매우 익숙하실 겁니다. 예를 들어 유튜브 앱을 실행해 보면, 비디오 목록이 보여지죠. 그 중 하나의 비디오를 선택하면 재생화면을 볼 수 있습니다. 그와 함께 아래에는 알고리즘이 우리를 새로운 비디오를 보도록 유혹하죠. 계속 화면을 이동하면서 보다가, 뒤로가기를 누르면 한 화면씩 이전에 봤던 화면을 보게 됩니다. 이것은 각 화면들을 Stack으로 관리하고 있기 때문이죠.

Android에서는 이것은 Activity Stack 이라고 부르고요, iOS에서는 이것은 Navigation Stack 이라고 부릅니다.

## 후위 표기 수식의 연산

Stack을 활용하는 사례를 하나 소개하겠습니다. 후위 표기법으로 표현된 수식의 연산인데요, 후위 표기법이란 연산자를 뒤에 기록하는 표기법입니다.

우리가 일반적으로 사용하는 수식은  $4 + 2$  처럼 연산자를 가운데 표기하는 중위 표시법입니다. 후위 표기법은  $4\ 2\ +$  처럼 수식을 뒤에 표기하는 방법입니다.

$2 + 2 / 2$  의 수식을 후위후식으로 표기하면  $2\ 2\ /\ 2\ +$  이렇게 될 겁니다. 이러한 수식을 Stack을 활용해서 풀이해 보면 이렇게 됩니다.

## 알고리즘

1. 수식에서 하나씩 꺼내어 Stack에 넣습니다.
2. 수식에서 꺼낸 것이 연산자인 경우 Stack에서 2개의 값을 pop 하여 계산을 수행하고 다시 push 합니다.
3. 수식의 모든 항목에 대해서 1, 2이 과정을 반복합니다.

## 수행과정

1. 처음 2를 꺼내어 스택에 넣습니다. (스택 : 2 )
2. 다음 2를 꺼내어 스택에 넣습니다. (스택 : 2 2 )
3. 다음에 수식에서 꺼낸 값이 연산자 / 이므로
4. Stack에서 두 개의 값을 pop 합니다. 그리고 두 값을 / 연산합니다. (연산 : 2 / 2)
5. 연산의 결과를 Stack에 push 합니다 . (스택 : 1 )
6. 수식에서 다음 항목인 2 를 Stack에 넣습니다. (스택 : 1 2 )
7. 수식의 다음 항목이 연산자 + 이므로
8. Stack에서 두 개의 값을 pop 합니다. 그리고 두 값을 + 연산합니다. (연산 : 1 + 2)
9. 연산의 결과를 Stack에 push 합니다 . (스택 : 3 )
10. 수식은 종료되고, Stack에 남은 값이 최종 답이 됩니다. (답 = 3)

## Queue는 어디에 사용될까?

### Event Queue

우리가 키보드를 타이핑 할 때, 마우스를 움직일 때, 화면을 클릭하고 프로그램을 수행할 때 이러한 모든 동작을 컴퓨터에서는 이벤트로 전달합니다. 운영체제는 사용자가 발생시킨 이러한 이벤트들을 계속 모아두었다가 순서대로 수행하죠. 이것을 Event Queue 라고 부릅니다.

우선순위나 인터럽트를 제외하고 모든 이벤트는 발생한 순서대로 동작합니다. 키보드를 타이핑 한대로 동작해야 하는 것이 당연하듯이죠.

컴퓨터에서는 굉장히 많은 이벤트가 발생하는데, 그것을 모두 바로바로 처리해줄 수 없는 상황일 때, 빠짐없이 실행해 줄 수 있도록 Queue에 쌓아두고 순차적으로 처리해 줍니다. 여기서 Queue는 Buffer의 역할을 해준다고 볼 수 있죠.

컴퓨터가 일시적으로 리소스의 부족 혹은 과부화로 인해 잠시 동작을 멈추는 현상을 Lack 이라고하죠. 보통 렉걸렸다고 합니다. 렉걸렸을 때 시간이 좀 지나면 그 사이에 우리가 클릭 하거나 타이핑 했던 것들이 한꺼번에 수행되는 것을 다들 경험해 보셨을 겁니다. 그 이벤트들이 그동안 Queue에 쌓여 있다가 한꺼번에 순서대로 수행되는 현상인 것입니다.

### Job Scheduler

역시나 Event Queue와 비슷한 얘기 입니다. 우리가 프린터로 문서를 출력할 때 여러장을 미리 프린터로 전송해 놓으면 프린터는 인쇄할 문서의 내용들을 미리 Spool 해두었다가 차례대로 출력합니다. 이것을 스푼기능이라고 하는데, 바로 Queue를 사용하는 것입니다.

또 스토어에서 혹은 인터넷에서 파일을 다운로드 할 때 여러 파일들을 순서대로 클릭해 놓으면 순서대로 다운로드가 수행되죠. 역시 Queue를 사용하고 있습니다. 이렇게 컴퓨터에서 이벤트나 수행 작업이나 모두 Queue를 활용해서 구성되고 동작하고 있습니다.