

# 큐 (Queue)

- 01 큐 이론
- 02 큐의 활용

신 제 용

# 01 큐 이론

제한된 기능을 제공하는 큐 추상자료형을 이해하고,  
이를 구현하는 자료구조를 학습합니다.

학습 키워드 - 큐, 접근 제한, 자료구조

Chapter 01  
큐 이론

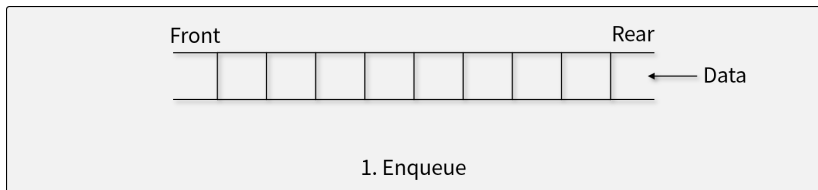
# 큐 (Queue)

- 스택과 유사하게 기능을 제한하는 추상자료형
- 선입선출 (First In First Out; FIFO)의 특성
- 자료가 입력된 순서대로 처리되어야 할 때 사용  
ex) 네트워크 트래픽, 프린터 버퍼 등

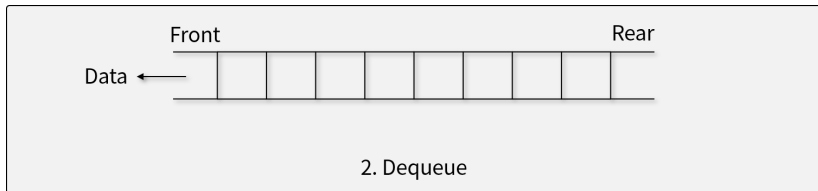
Chapter 01  
큐 이론

# 큐의 연산자

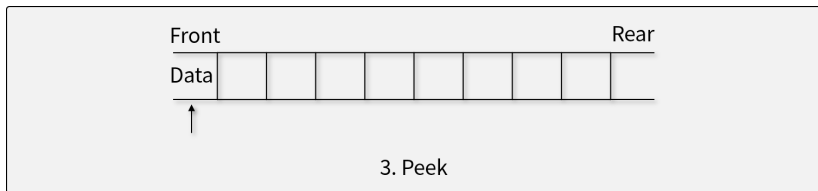
- 자료를 rear로 삽입하는 연산자 (put(), enqueue())



- 자료를 front에서 꺼내는 연산자 (get(), dequeue())

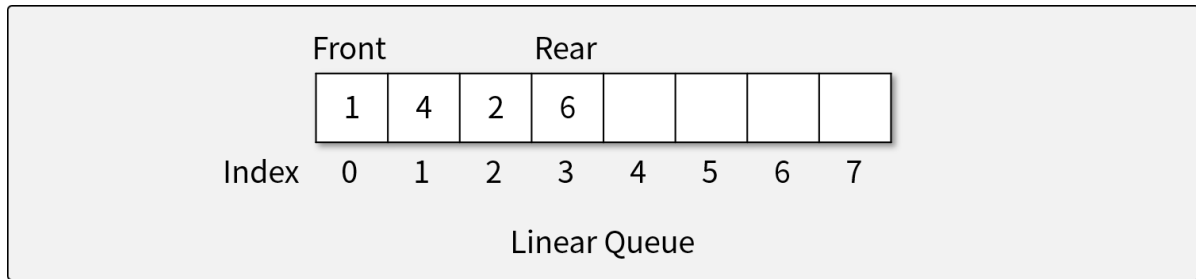


- front에 있는 자료를 반환하지만, 삭제하지는 않는 연산자 (peek())



# 큐의 구현 (1)

- 선형 큐 (Linear queue)
  - 배열을 이용하여 구현하는 가장 기본적인 큐
  - 한번 사용한 메모리 공간이 버려지는 문제
  - 반드시 오버플로우가 발생하는 구조적인 문제가 있다.

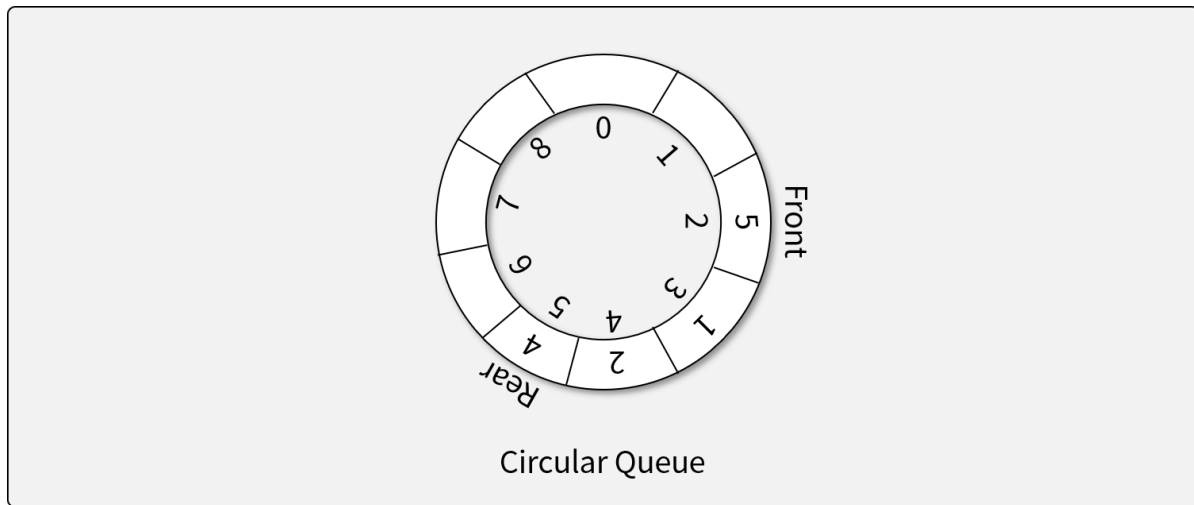


## Chapter 01

### 큐 이론

# 큐의 구현 (2)

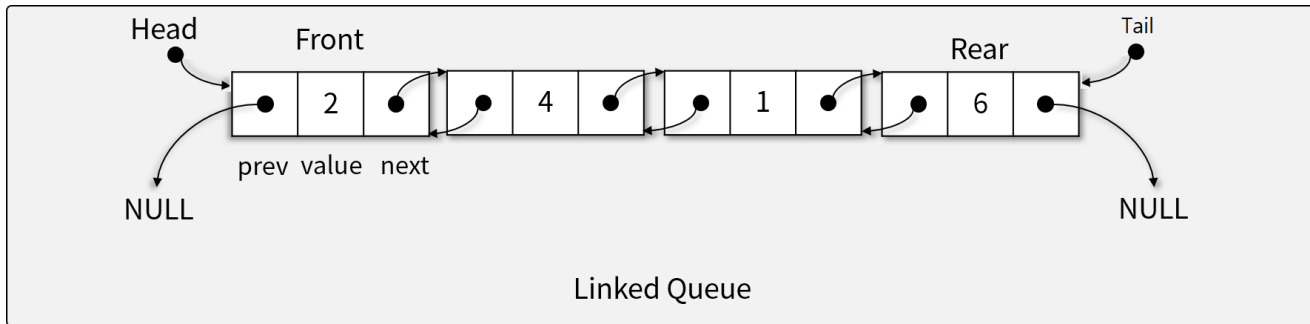
- 환형 큐 (Circular queue)
  - 선형 큐의 시작과 끝을 연결하여 메모리를 재활용하는 구조
  - 배열을 가득 채우기 전까지는 오버플로우가 발생하지 않는다.



Chapter 01  
큐 이론

# 큐의 구현 (3)

- 연결 리스트 큐 (Linked queue)
  - 연결 리스트를 이용해 큐 구현
  - 구조적으로 메모리 제약이 없기 때문에 많이 사용



Chapter 01  
큐 이론

# 02 큐의 활용

다음 챕터에서는 큐를 활용하는 예시를 확인하고 구현해봅니다.

Chapter 01

큐 이론



# 02 큐의 활용

큐를 활용하는 기초 유형 문제를 확인하고 직접 풀어봅시다.

학습 키워드 - 큐, 활용, 구현

Chapter 02  
큐의 활용

```
# Python에서 큐 사용하기 (리스트 응용)
```

```
queue = []  
for i in range(1, 11):  
    queue.append(i) # put()은 append()를 이용  
print(queue)  
  
print(queue[0]) # peek()는 [0]를 이용  
  
for i in range(1, 11):  
    val = queue.pop(0) # get()은 pop(0)을 이용  
    print(val, end=' ')  
print()
```

```
# Python 에서 큐 사용하기 (queue 패키지 사용)  
from queue import Queue
```

```
queue = Queue()  
for i in range(1, 11):  
    queue.put(i)
```

```
# print(queue.peek()) # peek()는 미구현
```

```
for i in range(1, 11):  
    val = queue.get()  
    print(val, end=' ')  
print()
```

## Chapter 02

### 큐의 활용

```
# Prob1.
# 카드 섞기 문제
# 1부터 N 까지의 번호로 구성된 N장의 카드가 있다.
# 1번 카드가 가장 위에 그리고 N번 카드는 가장 아래의 상태로 카드가 순서대로 쌓여있다.
# 아래의 동작을 카드 한 장만 남을 때까지 반복했을 때, 가장 마지막 남는 카드 번호를
# 출력하시오.
#
# 1. 가장 위의 카드는 버린다.
# 2. 그 다음 위의 카드는 쌓여 있는 카드의 가장 아래에 다시 넣는다.
#
# 예시 입력)
# N = 4
# 결과: 4
#
# N = 7
# 결과: 6
```

```
def solution(N):
    pass
```

```
if __name__ == '__main__':
    N = 4
    sol = solution(N)
    print(sol)
```

## Chapter 02

### 큐의 활용

```
# Prob2.
# 요세푸스 문제
# N과 K가 주어졌을 때 (N, K) 요세푸스 순열을 구하시오.
# N과 K는  $N \geq K$  를 만족하는 양의 정수이다.
# 1부터 N 번까지 N명이 순서대로 원을 이루어 모여 있다.
# 이 모임에서 원을 따라 순서대로 K번째 사람을 제외한다.
# 모든 사람이 제외될 때까지 반복하며 이 때, 제외되는 순서가 요세푸스 순열이다.
#
# 예시 입출력
# N = 5
# K = 2
# 결과: 2, 4, 1, 5, 3
#
# N = 7
# K = 3
# 결과: 3, 6, 2, 7, 5, 1, 4
```

```
def solution(N):
    pass

if __name__ == '__main__':
    N = 5
    K = 2
    sol = solution(N, K)
    print(sol)
```

## Chapter 02

### 큐의 활용