HAPPY HACKING

힙 (Heap), 셋 (Set)

HAPPY HACKING

1. **힙** (Heap)

2. **셋** (Set)

HAPPY HACKING

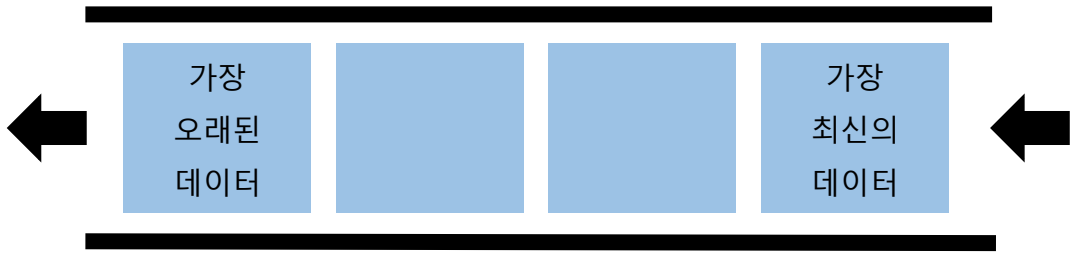
1. 힙 (Heap)

1. 힙 (Heap)

HAPPY HACKING

일반적인 큐(Queue)는 **순서를 기준**으로

가장 먼저 들어온 데이터가 가장 먼저 나가므로 FIFO(First-in First-out, 선입선출) 방식



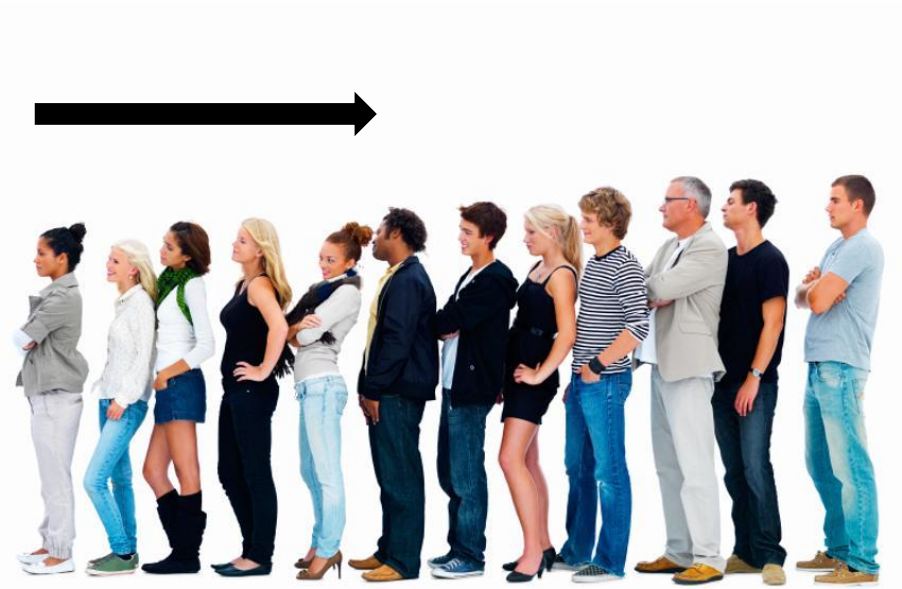
(일반) 큐의 동작 과정

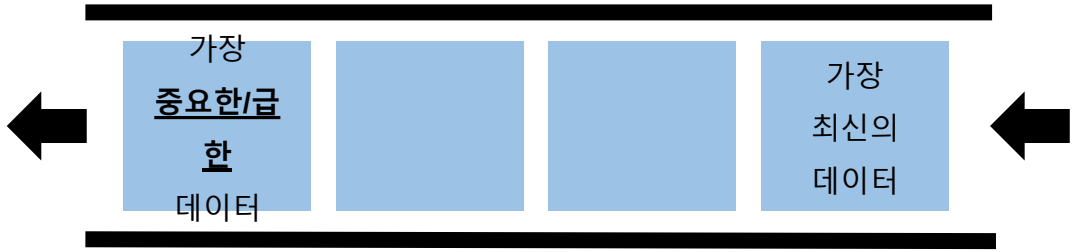
순서가 아닌 다른 기준으로는?

Heap이 필요한 경우 == Heap의 Use Case

1. 힙 (Heap)

우선순위 큐(Priority Queue)는 **우선순위(중요도, 크기 등 순서 이외의 기준)를 기준**으로
가장 우선순위가 높은 데이터가 가장 먼저 나가는 방식





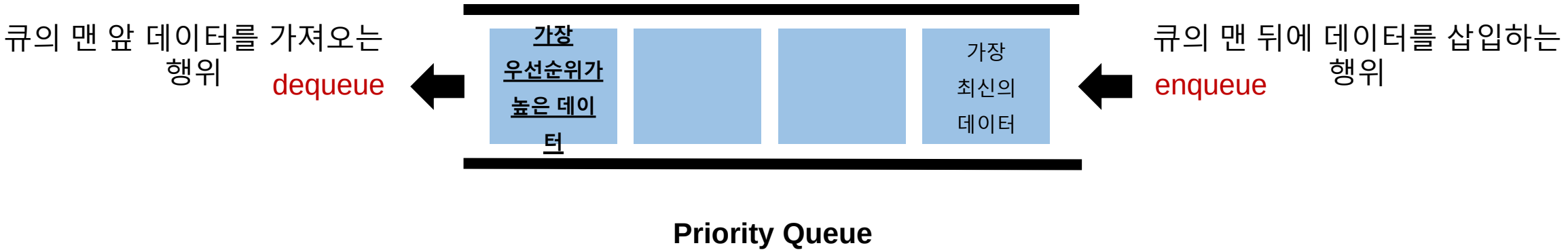
우선순위 큐의 동작 과정

우선순위 큐(Priority Queue)

순서가 아닌 우선순위를 기준으로 가져올 요소를 결정(dequeue)하는 큐

1. 가중치가 있는 데이터
2. 작업 스케줄링
3. 네트워크

1. 힙 (Heap)



Priority Queue

우선순위 큐(Priority Queue)를 구현하는 방법

1. 배열(Array)

2. 연결 리스트(Linked List)

3. 힙(Heap)

우선순위 큐 구현 별 시간 복잡도

연산 종류	Enqueue(추가)	Dequeue(
배열(Array)	O(1)	O(N)
(정렬된 배열)	O(N)	O(1)
연결리스트(Linked List)	O(1)	O(N)
(정렬된 연결리스트)	O(N)	O(1)
힙(Heap)	**O(logN)**	**O(logN)**

힙(Heap)의 특징

최대값 또는 최소값을 빠르게 찾아내도록 만들어진 데이터구조

완전 이진 트리의 형태로 느슨한 정렬 상태를 지속적으로 유지 한다.

힙 트리에서는 중복 값을 허용한다.

Heap은 언제 사용해야할까?

Heap이 필요한 경우 == Heap의 Use Case

Heap은 언제 사용해야할까?

1. 데이터가 지속적으로 정렬되야 하는 경우

2. 데이터에 삽입/삭제가 빈번할때

파이썬의 heapq 모듈

Minheap(최소 힙)으로 구현되어 있음(가장 작은 값이 먼저 옴)

삽입, 삭제, 수정, 조회 연산의 속도가 리스트보다 빠르다.

(배열, 연결리스트, 힙으로 구현 가능)

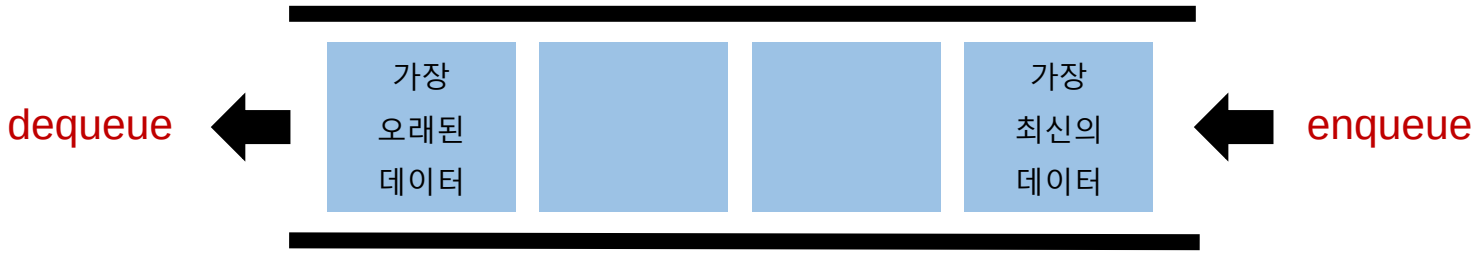
힙과 리스트 비교

연산 종류	힙(Heap)	리스트(List)
Get Item	O(1)	O(1)
Insert Item	O(logN)	O(1) 또는 O(N)
Delete Item	O(logN)	O(1) 또는 O(N)
Search Item	O(N)	O(N)

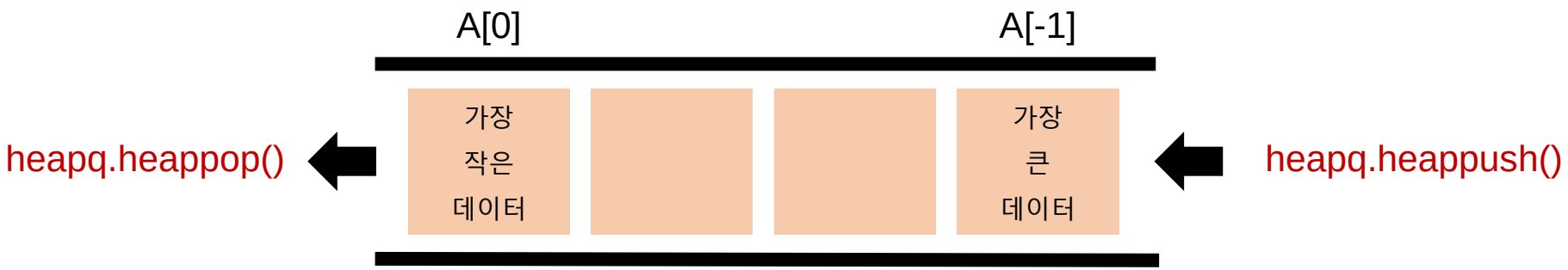
1. 힙 (Heap)

HAPPY HACKING

큐와 힙의 사용법 비교



Queue



Heap (min)

3. 딕셔너리 메서드

1) heapq.heapify( )
2) heapq.heappop(heap)
3) heapq.heappush(heap, item)

HAPPY HACKING

스택 연습

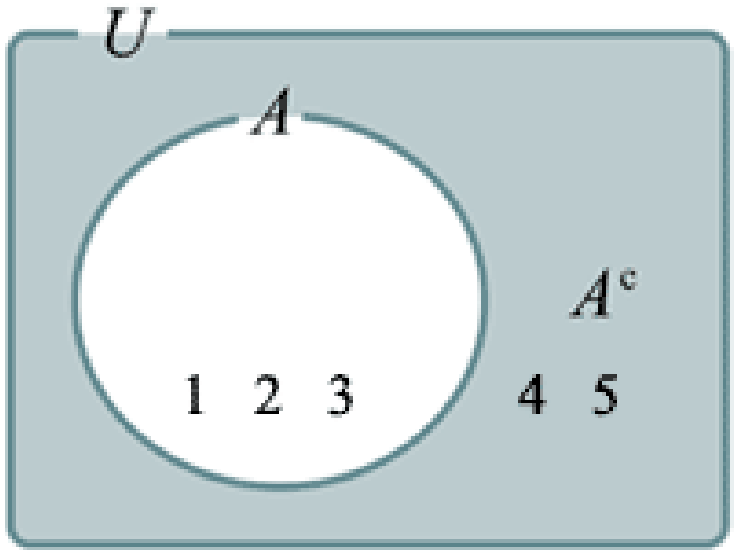
문제 번호	문제	링크
BOJ 1927	최소힙	https://www.acmicpc.net/problem/1927
BOJ 11286	절대값 힙	https://www.acmicpc.net/problem/11286

HAPPY HACKING

2. 셋 (Set)

2. 셋 (Set)

셋(set)은 수학에서의 '집합'을 나타내는 데이터 구조로 Python에서는 기본적으로 제공되는 데이터 구조이다.



$$A^c = \{x \mid x \in U \text{이고}, x \notin A\}$$

셋의 연산

1) .add( )
2) .remove( )
3) | (합)
4) - (차)
5) & (교)
6) ^ (대칭차)

Set은 언제 사용해야할까?

1. 데이터의 중복이 없어야 할 때 (고유값들로 이루어진 데이터가 필요할때)

2. 정수가 아닌 데이터의 삽입/삭제/탐색이 빈번히 필요할때

셋(Set) 연산의 시간 복잡도

연산 종류	시간복잡도
탐색	O(1)
제거	O(1)
합집합	O(N)
교집합	O(N)
차집합	O(N)
대칭 차집합	O(N)

HAPPY HACKING

셋 연습

문제 번호	문제	링크
BOJ 14425	문자열 집합	https://www.acmicpc.net/problem/14425
BOJ 1269	대칭 차집합	https://www.acmicpc.net/problem/1269