



Week1. Greedy Algorithm

기간 : 01/08(월) ~ 01/13(토)

▼ 관련 개념

- 선택의 순간마다 당장 눈앞에 보이는 최적의 상황만을 쫓아 최종적인 해답에 도달하는 방법
- 성립 조건
 - 탐욕적 선택 속성(Greedy Choice Property) : 앞의 선택이 이후의 선택에 영향을 주지 않는다.
 - 최적 부분 구조(Optimal Substructure) : 문제에 대한 최종 해결 방법은 부분 문제에 대한 최적 문제 해결 방법으로 구성된다.
- 해결 방법
 1. 선택 절차(Selection Procedure)
 - 현재 상태에서의 최적의 해답 선택
 2. 적절성 검사(Feasibility Check)
 - 선택된 해가 문제의 조건을 만족하는지 검사
 3. 해답 검사(Solution Check)
 - 원래의 문제가 해결되었는지 검사하고, 해결되지 않았다면 선택 절차로 돌아가 이전 과정 반복

▼ 문제 해결 과정에서 의문이 된 개념

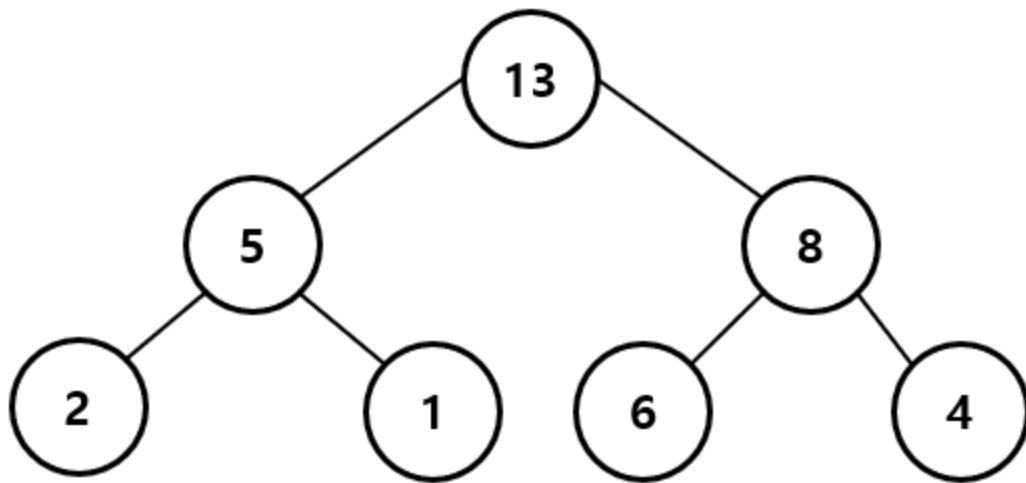
Priority Queue(우선순위 큐)와 Heap

- 기존 Queue : FIFO(First in-First Out)
- 우선순위 큐(Priority Queue) : 들어간 순서에 상관없이 우선순위가 높은 데이터가 먼저 나오는 Queue

- Heap으로 구현 가능
 - 삭제 및 삽입 과정에서 모두 부모와 자식 간의 비교만 이루어짐
 - 시간 복잡도가 배열, linked list로 구현할 때보다 낮음(삭제 : $O(\log 2n)$, 삽입 : $O(\log 2n)$)

Heap

- 완전 이진 트리(Complete Binary Tree)



- 모든 node에 지정된 값(우선순위)들은 자식 node들의 것보다 우선순위가 크거나 같다.
 - 직접 연결된 자식-부모 node 간의 크기만 비교하면 됨
 - heap으로 우선순위 큐를 구현할 경우, node에 저장된 값을 우선순위로 선정

▼ 공통 문제

- 백준-1715 : 카드 정렬하기
- 백준-1744 : 수 묶기
- 백준-1931 : 회의실 배정

▼ day1(01/09, 화)

1. 백준-1715 : 카드 정렬하기(미완)
2. 백준-11399 : ATM

▼ day2(01/10, 수)

1. 백준-1744 : 수 묶기

▼ day3(01/11, 목)

1. 백준-1931 : 회의실 배정

▼ day4(01/12, 금)

1. 백준-1715 : 카드 정렬하기
 - 일반적인 queue : FIFO(First in-First Out)
 - Priority Queue : 들어간 순서에 상관없이 우선순위가 높은 que