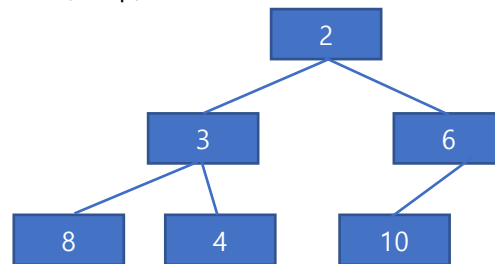


Week 9

- 주의 사항: 부정행위 금지(채점서버 외 인터넷 사용금지), STL 사용금지(string, vector는 사용 가능), **힙 기반**의 우선순위 큐를 구현하여 문제를 해결할것.
- 힙(heap) 생성시, upHeap을 이용하여 원소를 하나씩 힙에 삽입하는 방식으로 구현할 것.
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

문제 2 (Easy version)

자연수를 입력받아 배열(array) 기반의 힙(Heap)을 생성하여 입력된 숫자를 정렬하는 프로그램을 만들어보자.



예를 들어, 숫자가 8 4 6 2 3 10으로 입력이 되었을 때, min힙은 위의 그림과 같은 형태로 생성이 된다.
위 구조를 전체적으로 위쪽에서 아래쪽으로, 각 level의 왼쪽에서 오른쪽으로 읽으면 2 3 6 8 4 10이 되며,
숫자를 오름차순으로 정렬한 결과는 2 3 4 6 8 10이다.

cf. down heap을 하는 도중 heap order가 위반되었고, 자식들의 값이 동일할 경우, 왼쪽자식을 조절한다.

입력

1. 첫 번째 줄에 테스트 케이스의 개수 T 가 주어진다. ($1 \leq T \leq 20$).
2. 각 케이스마다 다음이 반복되어 나타난다.
 - 1) 각 케이스의 첫 번째 줄에 입력받을 숫자의 개수 N 과 p 가 주어진다. ($1 \leq p \leq N \leq 100,000$)
 - 2) 각 케이스의 두 번째 줄부터 N 개의 숫자가 입력된다. 입력되는 숫자는 1~20로 표현 가능한 정수이다.

출력

각 케이스 별로,

첫 번째 줄에는 최소힙(Min-Heap)을 전체적으로 위쪽에서 아래쪽으로, 각 level 별로 왼쪽에서 오른쪽으로 읽었을 때, p 번째 값을 출력한다.

두 번째 줄에는 정렬 기준에 따라 오름차순으로 정렬된 결과에서 p 번째 값을 출력한다.

예제 입출력

예제 입력	예제 출력
1	8 // 2 3 6 8 4 10
6 4	6 // 2 3 4 6 8 10
8 4 6 2 3 10	