

quick sort는 partition을 통해서 한 pivot을 가지고 양쪽으로 분류한 뒤 해당 pivot의 index를 리턴하는 방식이다. 본 과제에서는 재귀적으로 구현 했으며, pivot은 가장 마지막 요소로 삼는다.

quick sort는 주로 pivot이 얼마나 중앙에 있냐에 따라 성능이 결정된다. 본 과제에서는 맨 마지막 값을 넣었으므로, 성능은 input 값에 따라 달라진다.

partition하는 방식은. l, j값을 둔 뒤 j의 인덱스 값이 pivot보다 작으면 왼쪽, 크면 오른쪽에 두는 방식으로 구현한다. 이렇게 partition을 재귀적으로 작성하여 정렬을 한다.

위에 언급했다시피 pivot에 따라 알고리즘의 성능이 달라지므로, 최악의 경우에는 input에 따라 최악의 시간이 걸릴 가능성이 없지 않다. 이를 해결하고 어느 정도 평균의 성능을 발휘하기 위해 random값을 pivot으로 설정, 그리고 partition하는 방식이 바로 quick\_random이다.

quick\_random은 partition에서 원래 pivot을 맨 마지막으로 하는 반면, 입력을 p~r까지라고 할 때 하나를 고르는 방식이다.

알고리즘은 input에 따라 그 결과가 결정된다. 하지만 random pivot을 이용한다면 시간은 최선은 아니더라도 평균의 값을 가질거라고 예측한다.