Algorithm
Quick Sort

Bill Kim(김정훈) | ibillkim@gmail.com

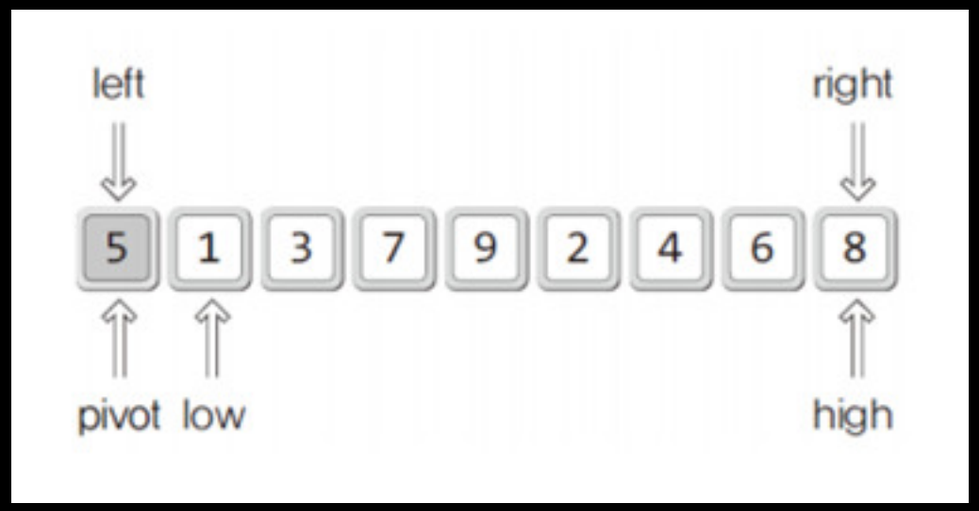
목차

Quick Sort

Quick Sort(퀵 정렬)는 대표적인 분할, 정복 정렬 알고리즘으로서 최악의 경우는 O(n^2)이지만 평균적으로는 O(nlogn)으로서 병합 정렬보다 보편적으로 빠른 속도를 가진 알고리즘입니다.

특정 값(Pivot)을 선택한 후 좌우로 작은 수와 큰 수 리스트를 나누어 최종 배열을 합치는 방법으로 정렬을 하는 방식입니다.

Pivot 값 선정에 따라서 일부 속도가 달라질 수 있는 그러한 알고리즘입니다.

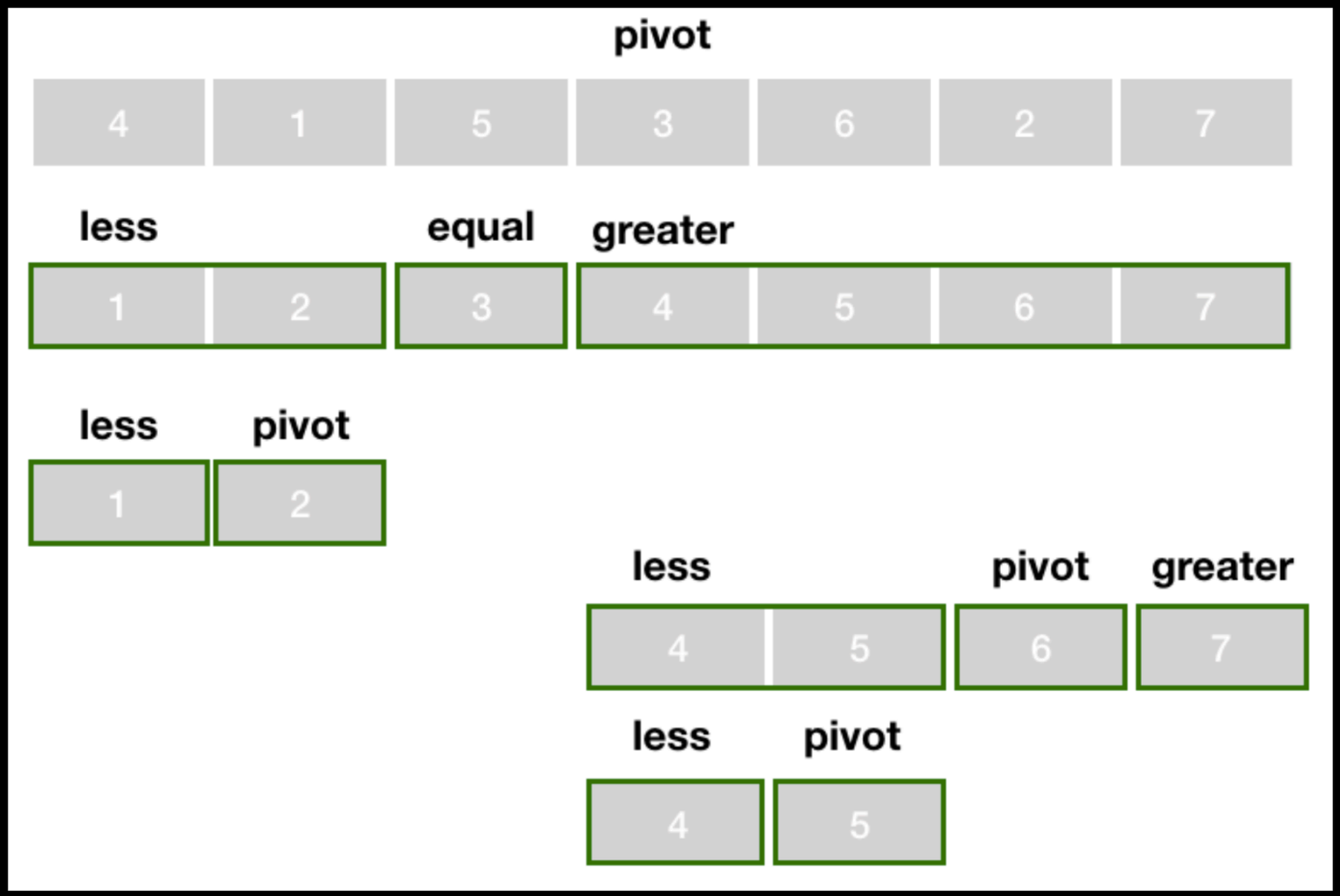


Concept

기본적인 알고리즘의 컨셉을 살펴보면 아래와 같습니다.

1. 기준 값(Pivot)를 하나 선정한다.
2. 해당 값을 중심으로 하여 작은 수와 큰 수로 나눈다.
   (L < 기준값 < R)
3. 재귀 호출을 통하여 마지막 배열의 크기가 1개일때까지 1~2번을 반복합니다.
4. 최종적으로 작은 수 배열과 기준 값(Pivot) 큰 수 배열을 합치면 정렬된 배열을 얻을 수 있습니다.

Concept


pivot
4
1
5
3
6
2
7
less
equal
greater
1
2
3
4
5
6
7
less
pivot
1
2
less
pivot
greater
4
5
6
7
less
pivot
4
5

Features

Quick Sort(퀵 정렬)는 아래와 같은 특징을 가진 알고리즘입니다.

1. 분할 정복을 활용한 빠른 정렬 알고리즘
2. 평균적으로 삽입이나 병합 정렬보다 빠른 속도를 가짐
3. 초기 Pivot 선택에 따라서 속도가 차이가 날 수 있다.
4. 평균은 O(nlogn), 최악의 경우 O(n^2)의 시간복잡도를 가짐
5. 최악의 시간 복잡도를 가지는 경우는 정렬되어 있거나 역순일 경우이므로 일반적인 상황에서는 거의 발생하지 않음

Implementation

Swift를 활용하여 퀵 정렬 알고리즘을 살펴보겠습니다.

```swift
func quickSort<T: Comparable>(_ array: [T]) -> [T] {
    if array.count < 2 {
        print("last array : \(array)")
        return array
    }
    else { print("array : \(array)") }

    let pivot = array.first!
    print("pivot : \(pivot)")

    let smaller = array.filter { $0 < pivot }
    let larger = array.filter { $0 > pivot }

    print("smaller : \(smaller)")
    print("larger : \(larger)")

    let result = quickSort(smaller) + [pivot] + quickSort(larger)

    print("smaller + pivot + larger : \(result)")

    return result
}
```

Implementation

```
let array = [3, 2, 5, 1, 4]

print(quickSort(array)) // [1, 2, 3, 4, 5]

// array : [3, 2, 5, 1, 4]
// pivot : 3
// smaller : [2, 1]
// larger : [5, 4]
// array : [2, 1]
// pivot : 2
// smaller : [1]
// larger : []
// last array : [1]
// last array : []
// smaller + pivot + larger : [1, 2]
// array : [5, 4]
// pivot : 5
// smaller : [4]
// larger : []
// last array : [4]
// last array : []
// smaller + pivot + larger : [4, 5]
// smaller + pivot + larger : [1, 2, 3, 4, 5]
// [1, 2, 3, 4, 5]
```

References

[1] Algorithm) 퀵 정렬(Quick Sort) in Swift : https://atelier-chez-moi.tistory.com/87

[2] 정렬 알고리즘 - Quick Sort (평균- nlogn, 최악- n^2) : https://zeddios.tistory.com/35

[3] [Swift]Quick Sort : http://minsone.github.io/programming/quick-sort-in-swift

[4] [알고리즘] 퀵 정렬 (Quick sort) : http://blog.naver.com/PostView.nhn?blogId=writer0713&logNo=221138306597&parentCategoryNo=&categoryNo=68&viewDate=&isShowPopularPosts=true&from=search

[5] 퀵 정렬 quick sort with swift : https://hyerios.tistory.com/70

References

[6] 알고리즘] 3. 퀵 정렬 : https://its-blog.tistory.com/m/105?category=801513

[7] 피벗설정에 따른 퀵정렬의 속도 : https://ghd5262.tistory.com/25

[8] [Swift 자료구조 ch11] 퀵소트 (Quick Sort) : https://kor45cw.tistory.com/247

[9] How do you implement Quick Sort in Swift? : https://medium.com/@notestomyself/how-do-you-implement-quick-sort-in-swift-d0dd0308a473

[10] Swift Quick Sort Algorithm With Recursion and Generics : https://medium.com/@augusteo/swift-quick-sort-algorithm-with-recursion-and-generics-78a256b3b392

Thank you!