

컴퓨터공학 All in One

C/C++ 문법, 자료구조 및 심화 프로젝트 (나동빈)
제 26강 - 퀵 정렬

학습 목표

퀵 정렬

- 1) 퀵 정렬의 원리를 이해하고 이를 C언어로 구현할 수 있습니다.

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

5	4	3	8	9	6	7	1	10	2
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

5	4	3	8	9	6	7	1	10	2
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

5	4	3	2	9	6	7	1	10	8
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

5	4	3	2	9	6	7	1	10	8
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

5	4	3	2	1	6	7	9	10	8
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

5	4	3	2	1	6	7	9	10	8
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

1	4	3	2	5	6	7	9	10	8
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

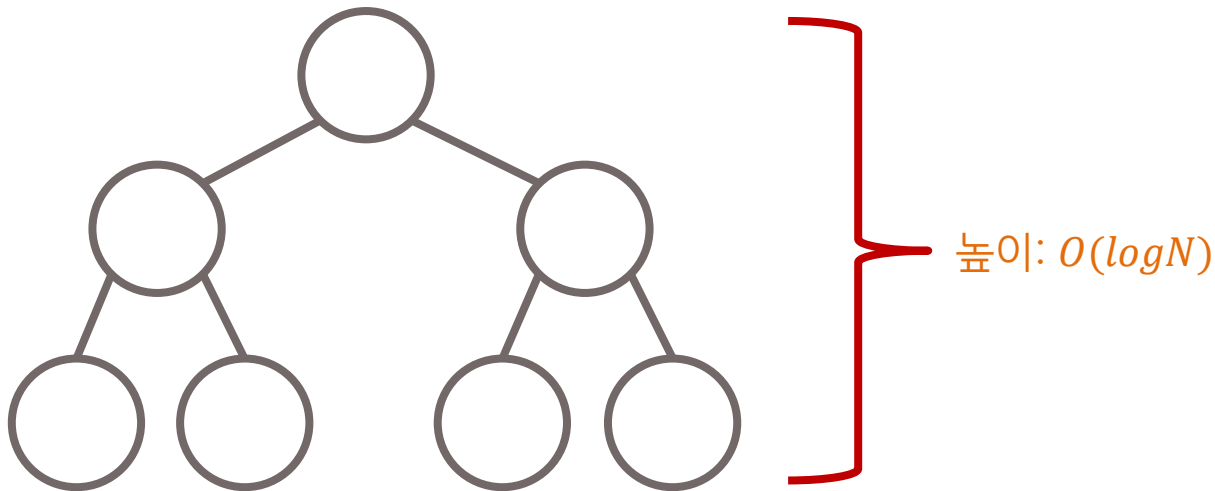
퀵 정렬이란 피벗을 기준으로 큰 값과 작은 값을 서로 교체하는 정렬 기법입니다. 값을 서로 교체하는 데에 N 번, 엇갈린 경우 교체 이후에 원소가 반으로 나누어지므로 전체 원소를 나누는 데에 평균적으로 $\log N$ 번이 소요되므로 평균적으로 $\theta(N \log N)$ 의 시간 복잡도를 가집니다.

1	4	3	2	5	6	7	9	10	8
---	---	---	---	---	---	---	---	----	---

퀵 정렬

퀵 정렬

원소를 절반씩 나눌 때 $\log N$ 의 시간 복잡도가 나오는 대표적인 예시는 완전 이진 트리입니다. 이러한 완전 이진 트리 형태는 흔히 컴퓨터 공학에서 가장 선호하는 이상적인 형태입니다.



퀵 정렬

퀵 정렬

퀵 정렬 - 배열 선언

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#define SIZE 1000

int a[SIZE];

int swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
```

퀵 정렬

퀵 정렬

퀵 정렬 - 퀵 정렬 구현하기

```
void quickSort(int start, int end) {  
    if (start >= end) return;  
    int key = start, i = start + 1, j = end, temp;  
    while (i <= j) { // 엇갈릴 때까지 반복합니다.  
        while (i <= end && a[i] <= a[key]) i++;  
        while (j > start && a[j] >= a[key]) j--;  
        if (i > j) swap(&a[key], &a[j]);  
        else swap(&a[i], &a[j]);  
    }  
    quickSort(start, j - 1);  
    quickSort(j + 1, end);  
}
```

퀵 정렬

퀵 정렬

퀵 정렬 - 퀵 정렬 사용해보기

```
int main(void) {  
    int n;  
    scanf("%d", &n);  
    for (int i = 0; i < n; i++) scanf("%d", &a[i]);  
    quickSort(0, n - 1);  
    for (int i = 0; i < n; i++) printf("%d ", a[i]);  
    system("pause");  
    return 0;  
}
```

퀵 정렬

퀵 정렬

퀵 정렬은 편향된 분할이 발생할 때 연산의 양이 $O(N^2)$ 입니다. 따라서 실제로 정렬을 함에 있어서는 퀵 정렬을 직접 구현하지 않습니다. 따라서 C++의 Algorithm 라이브러리를 사용합니다. Algorithm 라이브러리의 `sort()` 함수는 퀵 정렬을 기반으로 하되 $O(N\log N)$ 을 보장합니다.

배운 내용 정리하기

퀵 정렬

- 1) 퀵 정렬은 시간 복잡도가 $O(N\log N)$ 인 가장 보편적인 정렬 알고리즘입니다.