**<6장 정렬>**

-연속된 데이터를 기준에 따라서 정렬하기 위한 알고리즘이 정렬이다.

-정렬을 배워야 이진 탐색이 가능해짐! 선택 정렬, 삽입정렬, 퀵정렬, 계수 정렬을 다룬다(책에서)

**<선택 정렬>**

-컴퓨터가 데이터를 정렬할 때 가장 작은 것을 선택하는 방법을 선택 정렬 알고리즘이라 한다. 제일 작은 값을 계속 앞으로 보내다 보면 정렬이 된다.

-예제 코드

텍스트이(가) 표시된 사진

자동 생성된 설명

-파이썬 Swap 코드? 🡺 array[0], array[1] =array[1], array[0]을 하면 자동으로 둘의 값을 바꿔준다.

-선택 장렬은 기본 정렬 라이브러리(sort 같은)보다는 매우 비효율적 (시간이 많이 먹는다)이지만 특정 리스트에서 가장 작은 데이터를 찾는 경우가 많으므로 익숙해질 필요가 있다

**<삽입 정렬>**

-데이터를 하나씩 확인하며 데이터를 **적절한 위치에 삽입을 하는 정렬** 방법이다. 구현 난이도가 높지만 선택 정렬에 비해 실행 시간이 효율적이다. 🡺데이터가 거의 정렬되어 있을 때 효율적?

-데이터가 다른 곳으로 삽입될 때 **앞의 데이터는 이미 정렬되어 있다고 가정**하고 알고리즘을 작성.

\*\*뇌 피셜: 선택 정렬은 모든 리스트를 탐색해서 최소값을 “선택”하고 맨 앞에 두고 나서 계속해서 **모든 리스트를 순환하면서 그 중에서 작은 값을 앞으로 보내주는 방식**

삽입 정렬은 맨 앞에서부터 **한 칸씩 옆으로 가며 옆으로 갈 때 마다 앞에 있는 거랑 비교를 해서 작은 수를 앞으로 삽입시키는 개념이다**. 즉 모든 리스트를 계속해서 순환할 필요가 없기 때문에 시간이 절약된다. 그러니깐 **리스트 뒤로는 탐색을 안하고 현재 인덱스와 -1인덱스만 비교**를하면된다. 앞이 다 정렬 되어있다고 판단하고 앞을 기준으로 탐색!

**<퀵 정렬>**

-가장 많이 사용된다. **기준 데이터를 설정**하고 기준 데이터보다 큰 데이터와 작은 데이터의 위치를 변경하는 정렬이다.

기준 값을 정하고 기준 값보다 큰 수와 작은 수를 교환하고 리스트를 반으로 나눈다. 여기서 피벗(Pivot)을 활용한다. 피벗이 일종의 기준선이다.

과정

1)리스트에서 첫번째 데이터를 피벗으로 정한다.

2)피벗에서 오른쪽은 피벗보다 큰 숫자를, 리스트에서 왼쪽으로는 피벗보다 작은 숫자를 찾는다.

Ex🡺[5 7 9 0 3 1 6 2 4 8]이 있으면 5가 pivot이고 5에서 오른쪽으로는 5보다 큰 숫자를 리스트 맨끝인 8에서는 5보다 작은 수를 찾으면 7과 4가 나온다,

3)두 숫자를 바꿔준다. 그리고 다시 2번의 과정을 시작( 시작 점은 이전 진행 상황 부터이다!)

[5 4 9 0 3 1 6 2 7 8] 여기서는 9와 2가 나오게 된다 그럼 둘을 다시 바꿔준다.

[**5** 4 2 0 3 1 6 9 7 8]

4)진행 하다 왼쪽 진행도와 오른쪽 진행도가 엇갈린다? 🡺여기서는 1과 6에서 엇갈린다. 그러면 피벗을 제일 작은 숫자인 1과 위치를 바꿔준다

[1 4 2 0 3 **5** 6 9 7 8] 이러면 5를 기준으로 5보다 작거나 5보다 큰 숫자들로 분류가 된다.

5) 이 상태에서 5 기준으로 앞 뒤의 리스트를 재귀함수로 정렬 시켜 주면된다

<계수 정렬>

-계수 정렬은 특정한 조건이 부합할 때만 사용 가능하지만 매우 빠른 시간을 가지는 알고리즘이다.