# 코테 스터디

3주차 - 2024 / 05 / 14

### 원소 정렬하기

코딩 테스트에서의 정렬 알고리즘

• 원소를 특정 기준에 따라 순서대로 배열하는 것

9 3 6 8 1 2 4 7 5

9 3 6 8 1 2 4 7 5

9 3 6 8 1 2 4 7 5

\* The state of the state

• 보통 학부의 자료구조 / 알고리즘 에서 가장 많이 다루는 내용

Bubble Sort, Selection Sort, Insertion Sort, Merge Sort,
 Heap Sort, Quick Sort, ...

Bubble Sort, Selection Sort, Insertion Sort, Merge Sort,
 Heap Sort, Quick Sort, ...

• 이 스터디에서 정렬 알고리즘들의 상세 원리를 다루진 않을 예정

> 정렬 알고리즘이 궁금하면?

- 정렬 알고리즘 (나무위키)
- 정렬 알고리즘의 구현 (나무위키)

> 정렬이 필요할 때

- PS 환경에서 대부분의 경우 내장된 Sort 함수를 이용함
  - 대부분 sort 함수를 사용 못하는 상황일 때만 구현해서 사용

> 정렬이 필요할 때

- C++ : Sort()
- Python : {리스트 이름}.sort()
- Java : Arrays.sort()

• 상세한 내용은 각 언어별로 찾아보기

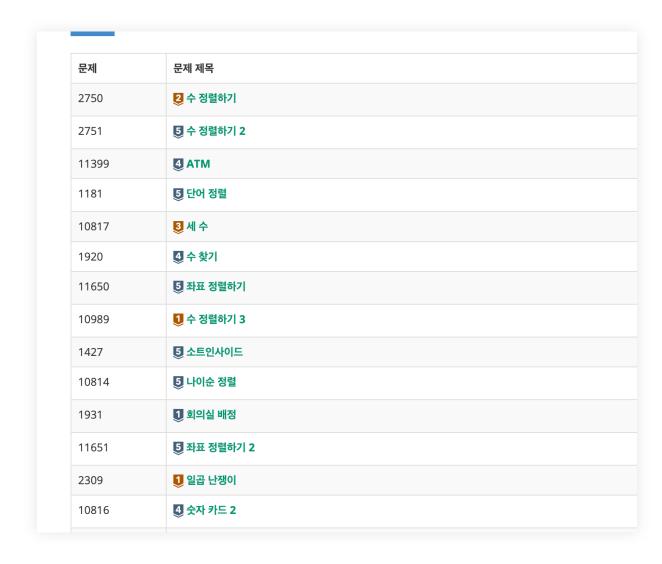
> 정렬이 필요할 때

• Sort() 구현체는 시간 복잡도  $O(N \log N)$ 으로 구현되어 있음

14

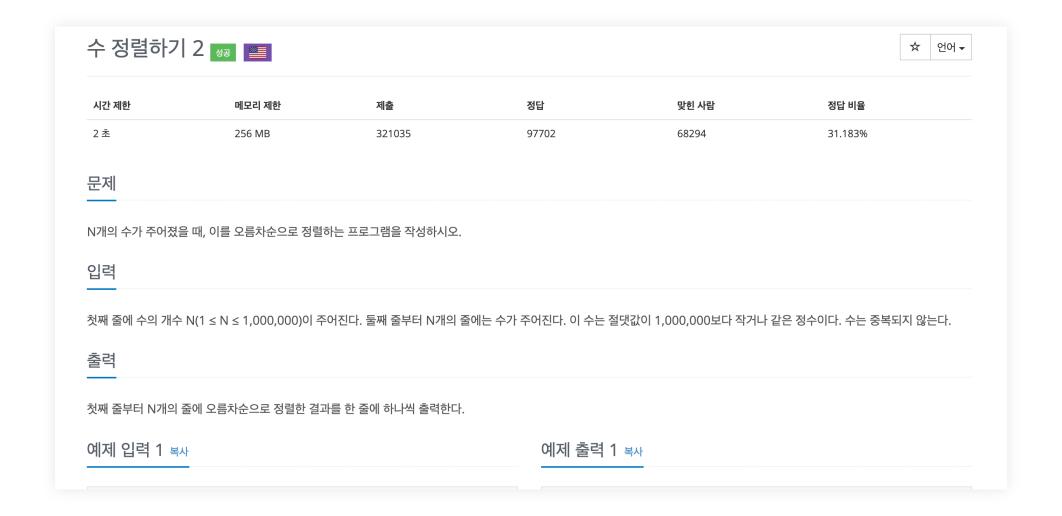
# 질문 시간

편하게 질문해주세요



백준 알고리즘 태그 '정렬'에
 연습할 수 있는 문제들이 많음

- BOJ 2751번 수 정렬하기 2
  - https://www.acmicpc.net/problem/2751



```
1 int main() {
     cin >> n;
     for (int i = 0; i < n; i++) cin >> v[i];
4
5
    // 각 언어 별로 sort함수 사용해보기
6
     sort(v, v + n);
     for (int i = 0; i < n; i++) cout << v[i] <<
9 }\n';
```

백준 2751

간단히 수를 정렬해보는 문제

수 정렬하기 2

백준 11650

좌표를 x좌표 기준으로 정렬해보는 문제

좌표 정렬하기

백준 10867

수를 정렬하고 중복을 제거해보는 문제

중복 빼고 정렬하기

# 원소 빠르게 찾아보기

이분 탐색 알고리즘

• 업-다운 게임 생각해보기

- 한 사람이 1~100 중 하나의 수를 생각함
- 다른 사람이 수 1개를 얘기함
- 그러면 대답으로 생각한 수보다 큰지 작은 지 얘기함

• 생각한 수를 맞춰보기

• 몇 번의 시도만에 찾을 수 있을까?

• 최대 7번 만에 찾을 수 있는 방법이 있음

- 가능한 수 범위에서 중앙값을 얘기함
- 업-다운을 들으면 불가능해지는 범위를 제외함
- 수가 확정될 때까지 이 과정을 반복하기

- 업-다운을 한 번 들을 때마다 수 범위가 절반으로 줄어듬
- 업-다운을 N번 들을 때마다 수 범위가  $\frac{1}{2^N}$ 으로 줄어듬

- 수 하나를 찾기 위해선  $log_2 N$  번을 해야함
- $6 < \log_2 100 < 7$ 이므로 최대 7번 물어보면 됨

> 그래서 이분 탐색이란

- 결정 문제의 답이 이분적일 때 사용할 수 있는 탐색 기법
- 자료가 정렬되어 있을 때만 사용할 수 있음

> 그래서 이분 탐색이란

• 앞에서 이야기한 업-다운 게임의 전략과 논리가 같음

## 수고하셨습니다

질문 받습니다