

코테 스터디

3주차 - 2024 / 05 / 14

원소 정렬하기

코딩 테스트에서의 정렬 알고리즘

> 정렬?

- 원소를 특정 기준에 따라 순서대로 배열하는 것

> 정렬?

9 3 6 8 1 2 4 7 5

> 정렬?

9 3 6 8 1 2 4 7 5



> 정렬?

9	3	6	8	1	2	4	7	5
				↓				
1	2	3	4	5	6	7	8	9

> 정렬 알고리즘의 종류

- 보통 학부의 자료구조 / 알고리즘 에서 가장 많이 다루는 내용

> 정렬 알고리즘의 종류

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

> 정렬 알고리즘의 종류

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

> 정렬 알고리즘의 종류

- 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

질문 시간

편하게 질문해주세요

> 정렬 연습하기

문제	문제 제목
2750	 수 정렬하기
2751	 수 정렬하기 2
11399	 ATM
1181	 단어 정렬
10817	 세 수
1920	 수 찾기
11650	 좌표 정렬하기
10989	 수 정렬하기 3
1427	 소트인사이드
10814	 나이순 정렬
1931	 회의실 배정
11651	 좌표 정렬하기 2
2309	 일곱 난쟁이
10816	 숫자 카드 2

- 백준 알고리즘 태그 '정렬'에
연습할 수 있는 문제들이 많음

> 정렬 연습하기

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

> 정렬 연습하기

수 정렬하기 2

성공



언어 ▼

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
2 초	256 MB	321035	97702	68294	31.183%

문제

N개의 수가 주어졌을 때, 이를 오름차순으로 정렬하는 프로그램을 작성하시오.

입력

첫째 줄에 수의 개수 N ($1 \leq N \leq 1,000,000$)이 주어진다. 둘째 줄부터 N개의 줄에는 수가 주어진다. 이 수는 절댓값이 1,000,000보다 작거나 같은 정수이다. 수는 중복되지 않는다.


출력

첫째 줄부터 N개의 줄에 오름차순으로 정렬한 결과를 한 줄에 하나씩 출력한다.

예제 입력 1 [복사](#)

예제 출력 1 [복사](#)

> 정렬 연습하기



```
1 int main() {  
2     cin >> n;  
3     for (int i = 0; i < n; i++) cin >> v[i];  
4  
5     // 각 언어 별로 sort함수 사용해보기  
6     sort(v, v + n);  
7  
8     for (int i = 0; i < n; i++) cout << v[i] <<  
9 }\\n';
```

> 정렬 연습하기

- **백준 2751**

간단히 수를 정렬해보는 문제

- **백준 11650**

좌표를 x좌표 기준으로 정렬해보는 문제

- **백준 10867**

수를 정렬하고 중복을 제거해보는 문제

수 정렬하기 2

좌표 정렬하기

중복 빼고 정렬하기

원소 빠르게 찾아보기

이분 탐색 알고리즘

> 이분 탐색이란

- 업-다운 게임 생각해보기

> 이분 탐색이란

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

> 이분 탐색이란

- 몇 번의 시도만에 찾을 수 있을까?

> 이분 탐색이란

- 최대 7번 만에 찾을 수 있는 방법이 있음

> 이분 탐색이란

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

> 이분 탐색이란

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

> 이분 탐색이란

- 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