

Data Structs-1

25NOVA Study

자료구조란

- 정의 : 데이터를 효율적으로 저장하고 관리하기 위한 구조
- 필요성:
 - 데이터 검색, 삽입, 삭제의 효율성 향상
 - 알고리즘 성능 최적화
 - 프로그램 구조화 및 유지보수 용이

진짜 왜 공부해야하나?

성적 관리...

1. 학생이 공부를 해야지

2. 회사에서는 니가 근면 성실하다는 **말**을 안믿음

1. 성적을 믿지

왜 배움?

알고리즘의 기초

- 알고리즘 수업 들어보면 *"이거 자료구조때 배운거 기억나지?"*
~~안나는데요~~

- 예시:

1. Dijkstra 알고리즘 → 우선순위 큐(Heap) 기반
2. DFS/BFS → 스택(Stack)과 큐(Queue) 사용
3. 머신러닝의 의사결정 트리 → 트리(Tree) 구조 활용

- 알고리즘을 잘 해야 코딩 테스트에 통과 가능
-> 돈벌기위해

왜 배움?

컴퓨터로 하는 일은 데이터량이 많다.

- 책 1권 : 자료구조 필요 없음
- 책 10권 : 배열
 - 가나다 순 정리
- 책 1,000,000권 : 효율적으로 찾고 관리하려면..
 - 해시 테이블?
 - 트리?
 - 그래프?

스터디의 목적

- 어차피 수업은 교수가 더 잘함
 - 시험도 교수가 낸다
 - 교수보다 더 잘 가르치는 강의 들으려면 돈 내고 들어...

스터디의 목적

- 혼자 공부하기 뻘셈
 - 강의 들어보니까 그렇더라
 - 내가 어디까지 아는건지
 - 그래도 해본사람은 좀 더 알지 않을까
 - 수업 들을때는 이해되는데 집에가면 뭘 말인지 모르겠어요

스터디의 목적

- 같이 실력을 키워 봅시다
- 스터디에서 방향성을 잡고
 - 배운 내용이 어디에 어떻게 쓰이는지 보고
 - 내 실력이 어느정도인가
 - 수업 들을때는 분명 아는거 같았는데
 - 코드로 짜보려니까 뭐가 뭔지 모르겠음
- 같이 연습 해 보아요
 - 공부는 결국 혼자 하는거
 - ~~니네 공부 니네가 알아서 하세요~~

자료구조 종류

- 배열(Array)
- 연결 리스트(Linked List)
- 스택(Stack)
- 큐(Queue)
- 트리(Tree) <- 스터디에서 이정도 하려나?
- 그래프(Graph)
- 해시 테이블(Hash Table)

스터디 계획

- 성능테스트 & C언어 복습(1)
- C언어 복습(2)
- 스택
- 트리
- 힙

C언어 배경지식

- 조건문 & 반복문
- 배열
- 포인터
- 동적할당
- 구조체

2750번

다 푼 사람은
2751번
풀어보세요

2750번

제출

맞힌 사람

숏코딩

재채점 결과

채점 현황

내 제출

난이도 기여

질문 게시판

수 정렬하기

성공

☆

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	128 MB	233051	134133	91856	58.280%

문제

N개의 수가 주어졌을 때, 이를 오름차순으로 정렬하는 프로그램을 작성하시오.

입력

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

출력

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


예제 입력 1 복사

5
5
2
3
4
1

예제 출력 1 복사

1
2
3
4
5

정답

```
1  #include <stdio.h>
2
3  void swap(int* x, int* y){
4      int tmp;
5      tmp = *x;
6      *x = *y;
7      *y = tmp;
8  }
9
10  int main() {
11     int n;
12     scanf("%d", &n);
13
14     int arr[1001];
15     for(int i=0; i<n; i++)
16         scanf("%d", &arr[i]);
```

```
18     for(int i=0; i<n-1; i++){
19         int min=arr[i];
20         int index=i;
21         for(int j=i+1; j<n; j++){
22             if(min>arr[j]){
23                 min = arr[j];
24                 index = j;
25             }
26         }
27
28         if(index>i)
29             swap(x: &arr[index], y: &arr[i]);
30     }
31     for(int i=0; i<n; i++)
32         printf("%d\n", arr[i]);
33 }
```

성능분석

- 우리가 작성한 코드가 얼마나 효율적인가?

공간 복잡도

- 프로그램이 실행될 때 요구되는 메모리 공간의 양

점근적 표기법 – Big-O notation

- $f(n)=O(g(n))$

- 이게 뭔데...

- $3n + 2 \rightarrow O(n)$

- $10n^2 + 4n + 2 \rightarrow O(n^2)$


- $6 \times 2^n + n^2 \rightarrow O(2^n)$

2. (10 points) 점근적표기(asymptotic notation) O (big O), Ω (omega), Θ (theta)를 그래프로 나타내고, 간략하게 설명하라.

3. (5 points) Big O로 표현할 수 있는 Time Complexity가 다음과 같을 때, reasonable한 time complexity를 선택하고 optimal한 순서로 정렬하라.

$O(n^2)$, $O(2^n)$, $O(m^n)$, $O(1)$, $O(n^5)$, $O(\log n)$, $O(n)$, $O(n \log n)$

한번 해 보세요

```
1  #include <stdio.h>
2
3  void swap(int* x, int* y){
4      int tmp;
5      tmp = *x;
6      *x = *y;
7      *y = tmp;
8  }
9
10  int main() {
11     int n;
12     scanf("%d", &n);
13
14     int arr[1001];
15     for(int i=0; i<n; i++)
16         scanf("%d", &arr[i]);
```

```
18     for(int i=0; i<n-1; i++){
19         int min=arr[i];
20         int index=i;
21         for(int j=i+1; j<n; j++){
22             if(min>arr[j]){
23                 min = arr[j];
24                 index = j;
25             }
26         }
27
28         if(index>i)
29             swap(&arr[index], &arr[i]);
30     }
31     for(int i=0; i<n; i++)
32         printf("%d\n", arr[i]);
33 }
```

$$3n^2 + 6n + 3 \rightarrow O(n^2)$$

2 2750번

제출

맞힌 사람

숏코딩

재채점 결과

채점 현황

내 제출

난이도 기여도

수 정렬하기 성공

시간 제한	메모리 제한	제출	정답
1 초	128 MB	233051	134133

문제

N개의 수가 주어졌을 때, 이를 오름차순으로 정렬하는 프로그램을 작성하시오.

입력

첫째 줄에 수의 개수 N ($1 \leq N \leq 1,000$)이 주어진다. 둘째 줄부터 N개의 줄에 수 N 개가 주어진다. 주어지는 수들은 $1,000$ 보다 작거나 같은 정수이다. 수는 중복되지 않는다.

수 정렬하기 2 성공

시간 제한	메모리 제한	제출	정답	맞힌 사람
2 초	256 MB	365593	113643	79667

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 1,000,000보다 작거나 같은 정수이다. 수는 중복되지 않는다.

```

1  #include <stdio.h>
2  #include <stdlib.h> // qsort 함수가 선언된 헤더 파일
3
4  int compare(const void *a, const void *b) // 오름차순 비교 함수 구현
5  {
6      int num1 = *(int *)a; // void 포인터를 int 포인터로 변환한 뒤 역참조하여 값을 가져옴
7      int num2 = *(int *)b; // void 포인터를 int 포인터로 변환한 뒤 역참조하여 값을 가져옴
8
9      if (num1 < num2) // a가 b보다 작을 때는
10         return -1;    // -1 반환
11
12     if (num1 > num2) // a가 b보다 클 때는
13         return 1;    // 1 반환
14
15     return 0; // a와 b가 같을 때는 0 반환
16 }

```

```

int main()
{
    int numArr[10] = { [0]: 8, [1]: 4, [2]: 2, [3]: 5, [4]: 3, [5]: 7, [6]: 10, [7]: 1, [8]: 6, [9]: 9 }; // 정렬되지 않은 배열

    // 정렬할 배열, 요소 개수, 요소 크기, 비교 함수를 넣어줌
    qsort( base: numArr, nel: sizeof(numArr) / sizeof(int), width: sizeof(int), compar: compare);

    for (int i = 0; i < 10; i++)
    {
        printf("%d ", numArr[i]); // 1 2 3 4 5 6 7 8 9 10
    }

    printf("\n");

    return 0;
}

```

5. (15 points) 행렬연산에 관한 함수이다. 행렬의 행과 열이 row와 col 매개변수로 전달될 때, 다음 함수를 완성하라.

```
/* malloc() 함수를 사용하여 matrix 자료구조 생성 */
int** malloc_matrix(int row, int col){
```

```
}
```

```
/* malloc_matrix()에서 할당된 메모리 해제
int free_matrix(int** matrix, int row, i
```

5. (15 points) 다음 프로그램에 대해 답변하라.

- (1) 행의 수가 row이고 열의 수가 col인 2차원 배열을 동적으로 할당는 코드를 완성하라.

```
int** create_matrix(int row, int col) {

    int** matrix = _____;

    for (int r = 0; r < row; r++) {

        _____;

    }

    return matrix;
}
```

C언어 복습 (1)

25NOVA Study

배열

배열

- 같은 자료형의 데이터를 여러 개 모아놓은 저장 공간
- 인덱스(index)는 0부터 시작
- 배열의 크기는 선언 후 변경 불가능

배열

기본 문법

```
자료형 배열이름 [크기];
```

예시

```
int arr[5] = {10, 20, 30, 40, 50};
```

구조체

구조체

- 서로 다른 자료형을 묶어서 하나의 데이터 타입으로 표현

구조체

구조체 선언 방법

```
struct 구조체이름 {  
    자료형 변수이름1;  
    자료형 변수이름2;  
    // 추가적인 변수  
};
```

구조체 사용 예시

```
// 구조체 정의  
typedef struct {  
    char name[20];  
    int age;  
    float height;  
} Person;  
  
// 구조체 변수 선언 및 초기화  
Person p1 = {"홍길동", 25, 175.5};  
  
// 데이터 접근  
printf("이름: %s, 나이: %d, 키: %.1f", p1.name,  
p1.age, p1.height);
```

포인터

포인터

- 메모리의 주소를 저장하는 변수

포인터

기본 문법

C ▾

자료형 *포인터이름;

📄 복사 캡션 ...

포인터 사용 예시

```
int num = 10;
int *ptr = &num; // num의 주소를 포인터 ptr에 저장

// 포인터를 통해 값 변경 가능
*ptr = 20;
printf("num: %d", num); // num: 20 출력
```

자료형	크기 (바이트)	표현 범위
char	1	-128 ~ 127 또는 0 ~ 255 (unsigned 사용 시)
short	2	-32,768 ~ 32,767
int	4	-2,147,483,648 ~ 2,147,483,647
long	4 또는 8	시스템에 따라 다름
long long	8	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
float	4	$\pm 3.4 \times 10^{-38} \sim \pm 3.4 \times 10^{38}$, 소수점 이하 6자리 정밀도
double	8	$\pm 1.7 \times 10^{-308} \sim \pm 1.7 \times 10^{308}$, 소수점 이하 15자리 정밀도
long double	8 또는 16	시스템에 따라 다름


```
typedef struct {  
    char a;  
    int b;  
} Example;
```

```
int a;
```

```
double b;
```

```
char c;
```

```
int *ptr1 = &a;
```

```
double *ptr2 = &b;
```

```
char *ptr3 = &c;
```

- 10818 최소 최대
- 2562 최댓값
- 10872 팩토리얼

다 푼 사람은 다음 문제도 풀어보세요

- 11720 숫자의 합
- 3052 나머지
- 10810 공 넣기
- 10872 팩토리얼