

과제: HW-1

- Code Test: <https://www.acmicpc.net/problem/2750>
- 해당 프로그램에 주석 (comment)를 한글로 단다. (추가할 수 있는 것은 최대한 추가)

#HW 1-1

i Insertion Sort 를 C 로 구현하고 제대로 작동되는지 문제풀이를 통해 살펴본다.

Requirements

- 첫째 줄에 수의 개수 $N(1 \leq N \leq 1,000)$ 이 주어진다. 둘째 줄부터 N 개의 줄에는 수 N 개가 주어진다. 이 수는 절댓값이 1,000 보다 작거나 같은 정수이다. 수는 중복되지 않는다.
- 첫째 줄부터 N 개의 줄에 오름차순으로 정렬한 결과를 한 줄에 하나씩 출력한다.

SOURCE CODE

```
#include <stdio.h>
/*
https://www.acmicpc.net/problem/2750
N 개의 수가 주어졌을 때, 이를 오름차순으로 정렬하는 프로그램을 작성하시오.
202033762 장민호
*/
#define MAX 1005 // N(1 ≤ N ≤ 1,000)
int arr[MAX];
void InsertionSort(int arr[], int n)
{
    int i, j;
    int insData; // 데이터 임시 보관소
    for (i = 1; i < n; i++)
    {
        insData = arr[i]; // i 번째 데이터를 insData 에 임시로 저장한다.
        for (j = i - 1; j >= 0; j--) // insData 위치 '전' 까지의 모든 숫자를 비교한다.
        {
            if (insData < arr[j]) // 오름차순 정렬. 즉, 만약 i 번째 데이터와, 그 전의
                // 데이터를 비교해서 그 전의 데이터가 더 클 경우, 그 두 데이터 위치를 바꿔야 오름차순이 유지될
                // 것이다.
            {
                arr[j + 1] = arr[j]; // 데이터 위치를 바꿔준다.
            }
            else
            {
                // 왜 break 인가? Insertion Sort 특성상, insData 위치 전까지의 수열은
                // 반드시 정렬된 상태이다. 따라서 insData 가 arr[j]보다 커지는 순간 insData 의 위치가 정해진
                // 것이고, 그 뒤는 이미 정렬되어 있으므로 더 이상 살펴볼 필요가 없다. 이러한 이유로 Insertion
                // Sort 는 이미 거의 정렬된 배열에 대해 매우 빠른 속도를 보여준다.
                break;
            }
        }
    }
}
```

```

    }
    }
    arr[j + 1] = insData; // 데이터 위치를 찾았으므로 삽입해준다.
}
}
void printArr(int N)
{
    for (int i = 0; i < N; i++)
    {
        printf("%d\n", arr[i]);
    }
}
void inputData(int N)
{
    for (int i = 0; i < N; i++)
    {
        scanf("%d", &arr[i]);
    }
}
int main()
{
    int N;
    scanf("%d", &N);
    inputData(N);
    InsertionSort(arr, N);
    printArr(N);
    return 0;
}

```

OUTPUT (Screen Shots)

#1 [제출 결과]

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이	제출한 시간
48756818	ahchjang	 2750	맞았습니다!!	1120 KB	0 ms	C99 / 수정	1856 B	25초 전

메모리의 경우, 애초에 데이터 양이 많지 않아 거의 차지하지 않았음을 볼 수 있다. 또한 걸린 시간의 경우 C 특성상 다른 언어들에 비해 개발시간은 길지만 작동시간은 짧기 때문에 Py3, Java 등의 언어에 비해 매우 빠른 속도를 보여준다.