



코딩과제 #28 (배점: 40점)

1. 설명:

삽입 정렬은 아직 정렬되지 않은 값을 이미 정렬된 배열 사이에 끼워 넣는 과정의 반복을 통해 정렬하는 방식이다. 예를 들어, '5, 3, 7, 2'이라는 리스트에서 3에 대하여 5보다 앞에 위치해야 하기 때문에 3을 해당 위치에 끼워 넣는다. 그러면 '3, 5, 7, 2'인 리스트가 만들어진다. 3과 5는 이미 정렬되었으므로 이번에는 7의 위치를 선정한다. 7은 3과 5보다 크기 때문에 교환이 이루어지지 않는다. 마지막으로 2에 대하여 2의 앞의 값들은 이미 정렬되어 있다. 2를 올바른 위치에 삽입하면 '2, 3, 5, 7'이라는 정렬된 리스트를 얻게 된다. 삽입 정렬을 위한 함수 `insertion_sort`를 완성하시오.

※ 유의사항: 본 코딩과제는 반드시 함수를 만들어야 함.

2. 출력:

오름차순 정렬이 완료된 8개의 자연수

3. 함수원형:

```
void insertion_sort(int numbers[], int len)
{

}
```

4. main 함수

```
#include <stdio.h>
#define ARR_SIZE 8

void main() {
    int numbers1[ARR_SIZE] = {9, 2, 5, 7, 4, 1, 1, 3};
    int numbers2[ARR_SIZE] = {2, 4, 9, 1, 4, 3, 5, 4};

    insertion_sort(numbers1, ARR_SIZE);
    insertion_sort(numbers2, ARR_SIZE);
}
```

5. 실행결과 예시:

[예시]

정렬 결과: 1 1 2 3 4 5 7 9

정렬 결과: 1 2 3 4 4 4 5 9