

# 제5장 정렬 프로그램

---

윈도우 프로그래밍 설계

청주대학교 전자공학과  
한철수

# 목차

---

- 정렬의 이해
- 거품 정렬 프로그램 작성 연습

# 정렬

- 정렬이란 나열되어 있는 자료를 일정한 순서대로 재배열하는 것을 말함.
  - 오름차순 정렬
    - 크기가 작은 값부터 큰 값순으로 정렬하는 것.
    - 예: (수) 1 3 5 7 9, (한글) 김유신 이순신 홍길동, (영어) apple bat cat
  - 내림차순 정렬
    - 크기가 큰 값부터 작은 값순으로 정렬하는 것.
    - 예: (수) 9 7 5 3 1, (한글) 홍길동 이순신 김유신, (영어) cat bat apple
- 대표적인 정렬 알고리즘
  - 거품 정렬(bubble sort)
  - 선택 정렬(selection sort)
  - 퀵 정렬(quick sort)
  - 힙 정렬(heap sort)

# 거품 정렬

- 거품 정렬은 앞에서부터 인접한 두 수를 비교해가면서 더 큰 수를 뒤에 배치시킴(오름차순 정렬의 경우). 이 과정이 한번 진행되고 나면 가장 큰 수가 맨 뒤에 위치하게 됨. 그 후, 마지막 칸을 제외하여 범위를 좁히고 위 과정을 반복함.
- 직관적이고 구현이 간단함.

다음과 같이 문자 'B', 'A', 'T', 'C', 'D'가 나열되어 있을 때, 알고리즘은 다음과 같다.

정렬 전	B	A	T	C	D	
정렬 과정	B	A	T	C	D	B와 A를 비교하여 교환한다.
	A	B	T	C	D	B와 T를 비교해서 교환한다.
	A	B	C	T	D	C와 T를 비교해서 교환한다.
	A	B	C	D	T	D와 T를 비교해서 교환한다.
정렬 후	A	B	C	D	T	

마지막 자리에 위치할 최대값을 구했음. 이제 마지막 칸을 제외하여 범위를 한 칸 좁히고 다시 비교 및 교환 과정을 반복함.

- 직접 해봅시다.
  - Z A C B D (오름차순 정렬)
  - 3 8 5 2 4 (내림차순 정렬)

# 거품 정렬 코드

```
for(int i=0; i<k-1; ++i){
    for (int j=0; j<k-1-i; ++j){
        if (nSource[j]>nSource[j+1]){
            temp=nSource[j];
            nSource[j]=nSource[j+1];
            nSource[j+1]=temp;
        }
    }
}
```

- nSource[k]: 자료가 저장된 배열
- k: 배열의 크기

현재 인덱스의 배열 값이 다음 인덱스의 배열 값보다 크면 두 값을 바꿈. (오름차순 정렬)

# 선택 정렬

- 범위내에서 가장 작은 수를 찾아 맨 앞자리 수와 교환한다(오름차순 정렬의 경우). 이제 첫 칸을 제외하여 범위를 좁히고 위 과정을 반복함.
- 직관적이고 구현이 간단함.

다음과 같이 숫자 35, 20, 90, 11, 10, 2가 나열되어 있을 때, 알고리즘은 다음과 같다.

정렬 전	35	20	90	11	10	2	
정렬과정	2	20	90	11	10	35	가장 작은 값 2를 1번째에 기억시킨다.
	2	10	90	11	20	35	2번째로 작은 값인 10을 2번째에 기억시킨다.
	2	10	11	90	20	35	3번째로 작은 값인 11을 3번째에 기억시킨다.
	2	10	11	20	90	35	4번째로 작은 값인 20을 4번째에 기억시킨다.
	2	10	11	20	35	90	5번째로 작은 값인 35를 5번째에 기억시킨다.
정렬 후	2	10	11	20	35	90	

- 직접 해봅시다.
  - 55 11 33 22 44 (오름차순 정렬)
  - 7 3 8 5 4 (내림차순 정렬)

# 선택 정렬 코드

```
void SelectSort(int data[], int n)
{
    for(int i=0; i<n-1; ++i){
        int min=data[i];           //범위의 첫 칸의 값을 최소값으로 설정
        int minIndex=i;          //범위의 첫 칸을 최소값 인덱스로 설정

        for(int j=i+1; j<n; ++j){
            if(min>data[j]){
                min=data[j];
                minIndex=j;
            }
        }
        data[minIndex]=data[i];
        data[i]=min;
    }
}
```

- data[]: 자료가 저장된 배열
- n: 배열의 크기

← 두번째 칸(i+1)부터 더 작은 수를 찾아 최소값과 인덱스를 갱신함.

← 첫 칸의 값을 찾아낸 최소값과 교환함.

# 거품 정렬 프로그램 작성 연습

- 만들고자 하는 프로그램

- 값을 입력하고 버튼을 누르면 거품 정렬을 수행하고, 결과를 아래에 표시함.

BubbleSort

입력

46 18 80 31

Bubble Sort

Bubble Sort 결과

18 31 46 80

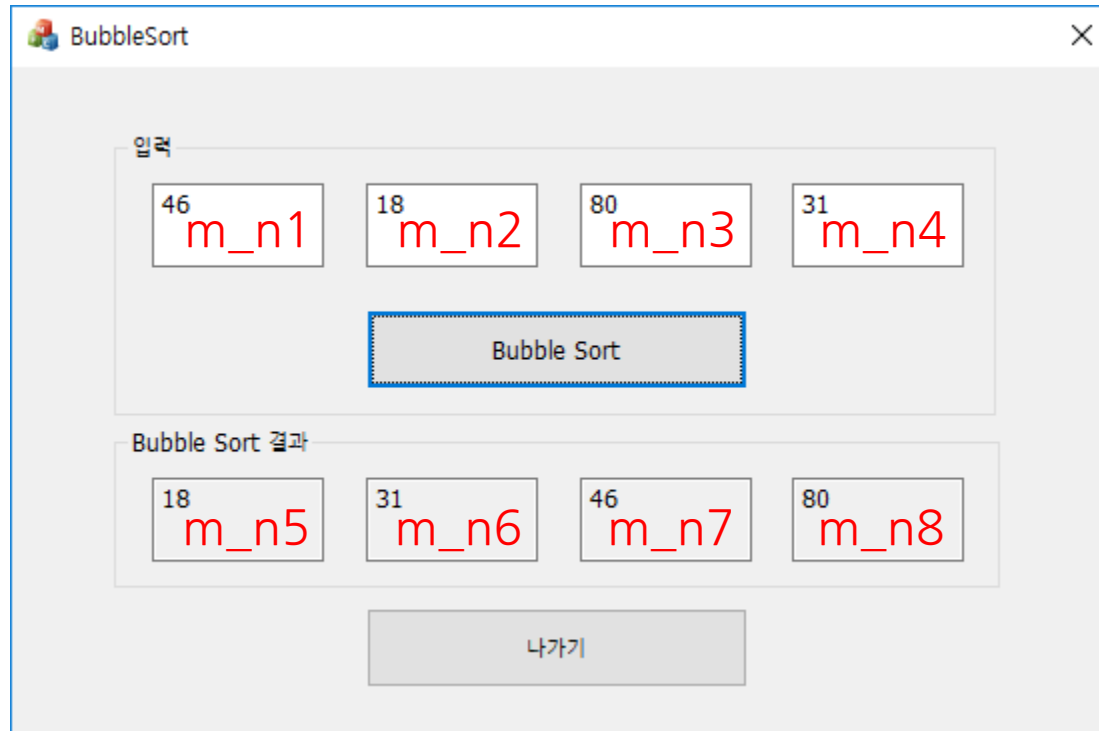
나가기

주의! 이번 장에서는  
에디트 컨트롤과  
연결시킨 변수의  
자료형이 int형임.  
4장에서는  
CString형이었음.



# 멤버 변수 추가

- 에디트 컨트롤에 변수를 연결함.
  - 변수의 자료형을 int형으로 할 것.



# 질문

---

**Q & A**