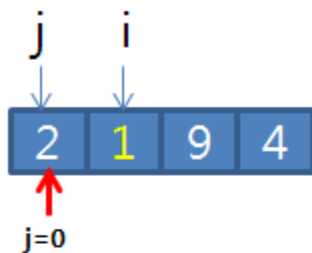


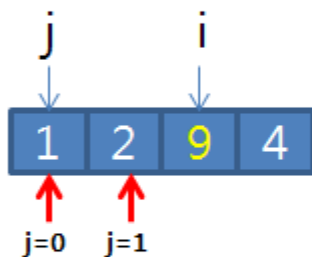
2. 버블정렬

버블정렬(Bubble sort)은 두 인접한 원소를 검사하여 정렬하는 방법이다. 시간 복잡도가 $O(n^2)$ 로 상당히 느리지만, 코드가 단순하기 때문에 자주 사용된다.

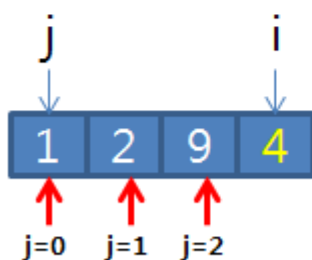
버블정렬(오름차순예)



[초기상태]루프 두 개가 필요하며 바깥쪽 루프는 i 값으로 안쪽 루프는 j 값으로 순회한다.
최초 i 값은 1부터 시작되며(i 가 0인 경우 좌측에 비교할 대상이 없다) j 값은 0부터 i 값보다 작을 때까지 순회하면서 j 가 가리키는 값이 크면 자리를 바꾼다.
2와 1을 비교하여 2가 크므로 자리 바꾼다.



i 값은 2가 되고 j 는 0부터 1까지 순회한다. j 가 0일때 즉 1과 9를 비교하면 정렬되어 있으므로 OK, $j=1$ 이 되어 2와 9를 비교한다. 역시 오름차순 정렬되어 있으므로 그대로 둔다.



다시 i 는 3이 되고 j 값은 0부터 2까지 순회한다. 먼저 1과 4를 비교하면 OK, 2와 4를 비교하면 역시 OK, 9와 4를 비교하면 9가 크므로 자리를 바꾼다.



i 값이 배열의 끝까지 도달하여 종료한다.

버블정렬 코드를 구현하고 테스트 하세요.

3. 선택정렬

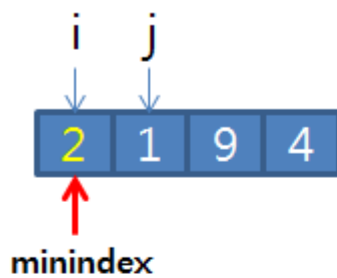
선택정렬은 키를 기준으로 나머지 값들 중 최소 또는 최대값을 선택하여 이 값을 키값과 바꾸면서 정렬하는 것이다.

[오름차순정렬의 예]

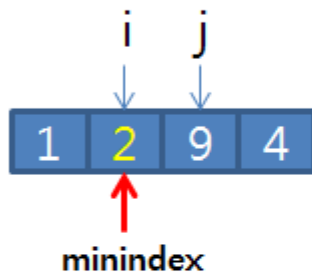
주어진 리스트 중에 최소값을 찾아서 그 값을 맨 앞에 위치한 값과 교체한다.

맨 처음 위치를 뺀 요소들에 대해 같은 방법으로 교체한다.

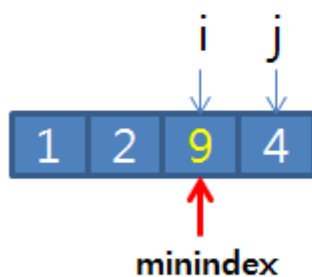
선택정렬(오름차순예)



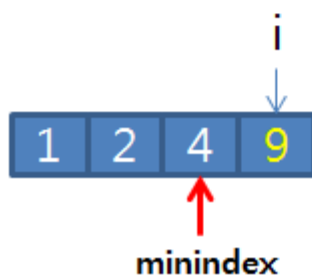
[초기상태]루프 두 개가 필요하며 바깥쪽 루프는 i값으로 안쪽 루프는 j값으로 순회한다. 최초 i값 0을 minindex로 두고 j가 1부터 3까지(1, 9, 4) 돌면서 i위치의 값과 비교하여 j 위치 값이 작다면 minindex= j 로 하여 minindex를 변경한다. j=1의 값이 최소값 1이므로 minindex = 1이 되고 이 값과 i번째 위치의 값을 바꾼다.



안쪽 for루프를 한번 순회했다면 아래처럼 값이 바뀌게 된다. 다시 i는 1이 되고 minindex를 1로 두고 안쪽 루프를 순회하여 9, 4는 2보다 크므로 minindex는 그대로 1이 된다. (위치 이동이 필요 없다.)



다시 i는 2가 되고 minindex를 2로 두고 안쪽 루프를 순회하여 4는 9보다 작으므로 minindex는 3이 되고 이 위치의 값 4와 9를 자리 바꾼다.



i위치가 3이 되면 종료되고, 배열은 정렬된다.

선택정렬 코드를 구현하고 테스트 하세요.

[LAB]두 수를 입력받아 사칙연산을 수행하는 프로그램을 작성하세요.

- switch case문 이용

- 두 수의 입력은 콤마로 구분해서 Console.ReadLine으로 받으세요.

[채워주세요]

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace 사칙연산
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("두수를 입력하세요.");
            Console.WriteLine("예 : 3,5");
            string str = Console.ReadLine();
            str.Trim();
            string[] strarr = str.Split(',');

            if (strarr.Length != 2)
            {
                Console.WriteLine("숫자2개를 콤마로 구분해서 입력하세요~");
                return;
            }

            int first, second;
            try
            {
                first = //여기 채워주세요 ;
                second = //여기 채워주세요 ;
            }
            catch
            {
                Console.WriteLine("입력이 잘못되었습니다.");
                return;
            }

            Console.WriteLine("{0} + {1} = {2}", first, second, Calc(first, second, "+"));
```

```
        Console.WriteLine("{0} - {1} = {2}", first, second, Calc(first, second, "-"));
        Console.WriteLine("{0} * {1} = {2}", first, second, Calc(first, second, "*"));
        Console.WriteLine("{0} / {1} = {2}", first, second, Calc(first, second, "/"));
    }

    static double Calc(int first, int second, string op)
    {
        double result = 0;
        switch (op)
        {
            //여기 채워주세요
        }
        return result;
    }
}
```