

0. ADT 구조체 타입 선언 : 모든 함수에 거쳐서 사용되어야 하기 때문에 전역 선언하였다.

1. input 함수 : 매개변수를 Point의 포인터형으로 지정하여 구조체 타입의 변수 x, y 의 주소값을 각각 받아 input 시 값이 input된 형태로 유지되도록 함

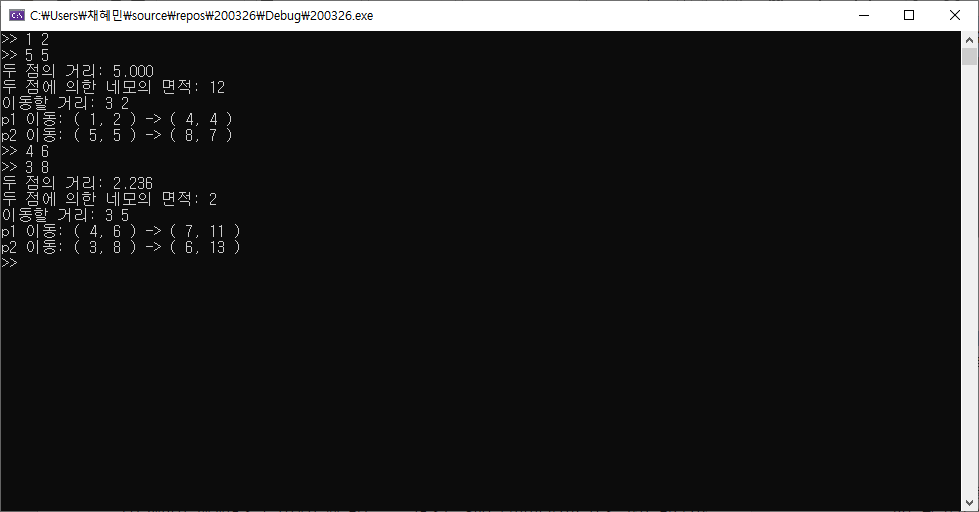
2. output 함수 : pt 의 x, y 값에 대하여 좌표형태로 출력되도록 함

3. distance 함수 : 좌표가 정수여도 sqrt 함수를 사용하기 때문에 거리 값이 실수일 수 있음을 감안하여 실수형으로 선언하였음. 지역변수 실수형 result에 sqrt식의 결과 값을 저장 후 result를 리턴해 주었음.

4. area 함수 : 두 좌표를 매개변수로 받아 조건문에서 두 좌표의 x 또는 y 값끼리 중 하나가 같으면 (|| 사용) 정의한 면적이 0이 되므로 0 리턴, 모두 같지 않은 경우는 math.h의 abs 함수(절댓값)를 이용해 면적(int)을 반환하였다.

5. equals 함수 : 두 좌표를 매개변수로 받아 조건문에서 두 좌표의 x, y 끼리 모두 같으면 (&& 사용) 1, 그렇지 않으면 0을 반환.

6. move 함수 : test 함수에서 좌표 pt2를 input 받아 기존 pt1의 좌표에 더하여 값을 바꾸는 거이므로 pt1은 포인터 변수로 선언, x, y를 바꾸어주었다.



후기

원하는 타입의 변수를 구조체로 묶고, 또 그 구조체를 타입 선언하여 변수형처럼 사용할 수 있는 것이 매우 편리하였다.

다만 Point의 변수 x, y를 입력(scanf\_s) 받을 때나 수정할 때 일일이 진행해주어야 한다는 점과

pt.x 또는 포인터 선언했을 때는 pt->x 와 같이 변수를 참조해야 한다는 것이 익숙하지 않았다.

특히 입력(scanf\_s) 받을 때는 변수 선언 시 초기화 하는 방법과 같이 진행할 수 없을까? 라는 의문이 생겨 \*pt ={ x, y } 와 같이 초기화 해보았으나 유효하지 않은 식이라고 오류가 떴다. 이유는 잘 모르겠다.