

1. 복소수를 나타내는 구조체 `Complex`를 정의하고, 두 복소수 `p`, `q`를 전달받아 이들의 합 `r`을 구하여 리턴하는 함수 `void Add(struct Complex* p, struct Complex* q, struct Complex* r)`, 두 복소수 `p`, `q`를 전달받아 이들의 곱 `r`을 구하여 리턴하는 함수 `void Multiply(struct Complex* p, struct Complex* q, struct Complex* r)`을 작성하시오. 또한 이를 이용하여 사용자로부터 두 복소수 값을 입력 받아, 그들의 합과 곱을 출력하는 프로그램을 작성하시오.  
(Ex. (1.0 -1) (2 3.0)  $\rightarrow$  (3 2) (5 1))  
(\*) 입력은 (실수부 허수부)의 형식이다. 즉, (1.0 -1)은  $1-i$ 를 의미하고, (2 3.0)은  $2+3i$ 를 의미한다.
2. 4차원 벡터를 나타내는 구조체 `Vector`를 정의하고, 두 벡터 `p`, `q`를 전달받아 이들의 합 `r`을 구하여 리턴하는 함수 `void Add(struct Vector* p, struct Vector* q, struct Vector* r)`, 벡터 `p`와 실수 `q`를 전달받아 이들의 곱 `r`을 구하여 리턴하는 함수 `void Multiply(struct Vector* p, double q, struct Vector* r)`, 두 벡터 `p`, `q`를 전달받아 이들의 dot product 값을 구하여 리턴하는 함수 `double DotProduct(struct Vector* p, struct Vector* q)`를 각각 작성하시오. 또한 이를 이용하여 두 개의 4 차원 벡터를 입력 받아 두 벡터의 합, 차, dot product 값을 출력하는 프로그램을 작성하시오. 단 벡터의 element의 값은 실수로 주어진다.  
(Ex. [1 2 3 4] [1 -1 1 -1]  $\rightarrow$  [2 1 4 3] [0 3 2 5] -2)
3. 선분은 그 선분의 두 개의 끝점으로 표현할 수 있다. 2차원 평면상의 좌표를 나타내는 구조체 `Point`를 정의하고, 이를 이용하여 두 점으로 선분을 나타내는 구조체 `Segment`을 정의하시오. 또한 선분 `seg`를 전달받아 선분의 기울기 `s`와 y-절편 `y`를 구하여 리턴하는 함수 `void LineParameter(struct Segment* seg, double* s, double* y)`와, 선분 `seg`를 전달받아 직선의 중점 `c`를 구하는 함수 `void Middle(struct Segment* seg, struct Point* c)`를 각각 작성하시오. 또한 이를 이용하여 선분의 두 끝점을 입력 받아 선분의 기울기, y-절편, 중점을 출력하는 프로그램을 작성하시오. 단 좌표는 실수로 주어지며 선분의 두 끝점의 x-좌표값은 서로 다르다.  
(Ex. (1 1) (2.0 3.0)  $\rightarrow$  2 -1 (1.5, 2))
4. 원점 좌표 (0, 0)이 입력될 때까지 X-Y 평면상의 좌표점들을 입력 받은 후에 원점과의 거리가 가장 가까운 좌표를 출력하는 프로그램을 작성하시오. 단, 좌표값은 정수인 것으로 가정하고 각 좌표는 구조체 `Point`로 표현하시오. 또한 좌표 구조체 `p`를 전달하면 그 좌표와 원점과의 거리를 리턴하는 함수 `double Distance(struct Point* p)`를 작성하고 이를 이용하여 프로그램

을 작성하시오. 단 거리가 같은 좌표가 여럿인 경우 먼저 입력된 좌표를 출력하시오.

(Ex. (1 1) (-4 -2) (-1 1) (2 3) (0 0) → (1 1) / (1 2) (-1 1) (4 5) (2 3) (0 0) → (-1 1))

5. 원점 좌표 (0, 0)이 입력될 때까지 X-Y 평면상의 좌표들을 입력 받은 후에 또 하나의 좌표 (x, y)를 더 입력 받아 좌표 (x, y) 가 앞서 입력한 좌표들 중에 있으면 "Yes", 없으면 "No"를 출력하는 프로그램을 작성하시오. 단 처음 입력되는 좌표의 수는 5개 이하로 가정하고 좌표 값은 정수이며, 각 좌표는 구조체 Point로 표현하시오.

(Ex. (4 5) (-4 -2) (-1 1) (2 3) (0 0) (3 3) → No / (4 5) (-1 1) (2 3) (0 0) (-1 1) → Yes)