**Lab 11. Structure #3**

2018/01/11

1. 2차원 평면에 존재하는 점과 직선에 대하여 다음의 조건을 만족하는 프로그램을 작성하세요.

★ 프로그램 설명

■ 2차원 평면에 존재하는 점과 직선은 double형 배열의 자료형을 재정의해서 사용할 수 있다.

typedef double Point2D[2]; // 점은 (x,y)로 표현하며, [0]: x, [1]: y가 저장됨.

typedef double Line2D[3]; // 직선은 ax+by+c=0으로 표현하며, [0]: a, [1]: b, [2]: c가 저장됨

■ 두 점을 입력(pt1, pt2)으로 두 점을 지나는 직선의 방정식 l을 계산하는 함수를 작성한다.

void calLineEquation(Point2D pt1, Point2D pt2, Line2D l);

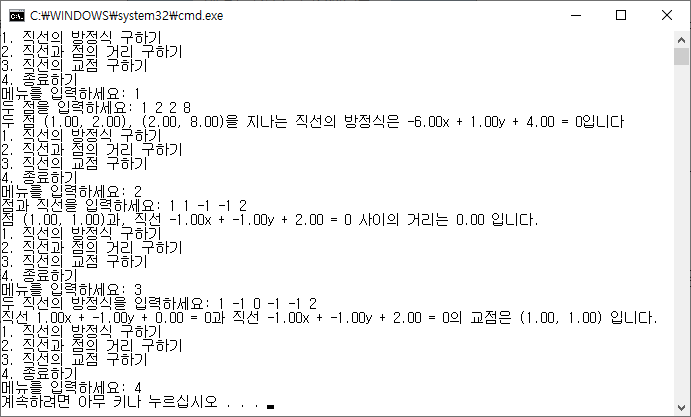
■ 한 점 pt와 직선의 방정식 l이 주어지면 점과 직선사이의 거리를 계산하여 반환하는 함수를 작성한다.

double getDistance(Point2D pt, Line2D l);

■ 두 직선의 방정식 l1과 l2가 주어지면, 두 직선의 교점을 계산하여 pt에 저장하는 함수를 작성한다.

void calIntersectionPoint(Line2D l1, Line2D l2, Point2D pt);

★ 실행 예



2. 3차원 평면에 존재하는 벡터에 대하여 다음의 조건을 만족하는 프로그램을 작성하세요

★ 프로그램 설명

■ 3차원 평면에 존재하는 벡터는 double형 배열의 자료형을 재정의해서 사용할 수 있다.

typedef double Vec3D[3]; // 벡터는 [x, y, z]로 표현하며, [0]: x, [1]: y, [2]: z가 저장됨.

■ 두 벡터 v1과 v2가 주어졌을 때, 동일한 벡터이면 1을 그렇지 않으면 0을 반환하는 함수를 작성한다.

int isEqualVec (Vec3D v1, Vec3D v2);

■ 벡터 v의 크기를 반환하는 함수를 작성한다.

double norm (Vec3D v);

■ 두 벡터 v1과 v2의 내적을 반환하는 함수를 작성한다.

double inner(Vec3D v1, Vec3D v2);

■ 두 벡터 v1과 v2의 내각을 반환하는 함수를 작성한다. 이 때 내각은 radian이 아닌 degree를반환해야 한다.

double angle(Vec3D v1, Vec3D v2);

■ 벡터 v1을 벡터 v2에 투영한 벡터를 v3에 저장하는 함수를 작성한다.

void project(Vec3D v1, Vec3D v2, Vec3D v3);

■ 두 벡터 v1과 v2의 합을 벡터 v3에 저장하는 함수를 작성한다.

void addVec(Vec3D v1, Vec3D v2, Vec3D v3);

■ 벡터 v1의 스칼라 곱을 벡터 v2에 저장하는 함수를 작성한다.

void multScalarToVec(Vec3D v1, double scalar, Vec3D v2);

★ 실행 예

