

## EJERCICIO DE TIPOS DE DATOS

Haciendo uso de WxDevC++ / Code Blocks hay que definir unos tipos de dato que sirvan de soporte básico al programador de otras librerías o aplicaciones en las que la geometría 3D juega un papel importante.

El motor geométrico constará de los elementos matemáticos más básicos de la geometría punto-vector-recta-plano:

**TD Punto.** Se trata del tipo más básico. Representa un punto en el espacio 3D.

### Propiedades:

- Coordenadas X, Y, Z de tipo real.

### Funciones:

- Crear (sin parámetros):  $X=Y=Z=0$ .
- Crear a partir de los valores de sus tres coordenadas.
- Obtener una de sus componentes: obtenerX(), obtenerY(), obtenerZ().
- Distancia euclídea a otro punto:

$$D(p1(x1,y1,z1),p2(x2,y2,z2)) = \sqrt{(x1-x2)^2+(y1-y2)^2+(z1-z2)^2}$$

**TD Vector.** Representa una dirección en el espacio (centrada en el origen de coordenadas).

### Propiedades:

- Coordenadas X, Y, Z de tipo real.

### Funciones:

- Crear (sin parámetros):  $X=Y=Z=0$ .
- Crear a partir de los valores de sus tres coordenadas.
- Crear a partir de dos puntos.

-Obtener alguna de sus direcciones: obtenerX(), obtenerY(), obtenerZ().

-Norma o módulo del vector: modulo().

$$|v| = \sqrt{x^2 + y^2 + z^2}$$

-Normalizar Vector: Modificar las coordenadas x,y y z para que el módulo sea 1. (esto se consigue dividiendo el valor de cada coordenada del vector por el módulo)

-Producto escalar con otro vector. Devuelve un escalar.

$$v1(x1,y1,z1) * v2(x2,y2,z2) = x1 * x2 + y1 * y2 + z1 * z2$$

-Producto vectorial con otro vector.

-Producto de un vector por un escalar. Devuelve un nuevo vector.

-Perpendicular(Vector v2). Obtiene el vector perpendicular al actual y a v2. (Se obtiene a partir del producto vectorial de los dos vectores).

-Sumar vectores. Devuelve un nuevo vector.

-Multiplicar un vector y un escalar. Devuelve un nuevo vector.

**TD Recta.** Representa el lugar geométrico de una recta en el espacio.

#### Propiedades:

-Se puede almacenar de varias formas: como dos puntos o como un punto y un vector dirección. Todo depende de la ecuación que se use. Preferimos la representación punto-vector por su similitud con un plano. Al punto lo llamaremos P y al vector V.

#### Funciones:

-Crear (sin parámetros): P=(0,0,0), V=(1,0,0)

-Crear a partir de dos puntos (necesario calcular el vector dirección que une los dos puntos).

-Crear a partir de un punto y un vector.

-Obtener alguna de sus componentes: obtenerPunto y obtenerVector.

-Distancia a un punto.

-Distancia a otra recta.

-Posición relativa con otra recta: hay cuatro casos: coincidentes, paralelas, cruce o secantes.

**TD Plano.** Representa el lugar geométrico de un plano infinito en el espacio. Se define a partir de un punto y un vector perpendicular a dicho plano (aunque son posibles otras representaciones).

Propiedades:

-Punto P y vector normal V (igual que la clase recta).

Funciones:

-Crear (sin parámetros):  $P=(0,0,0)$ ,  $V=(1,0,0)$

-Crear a partir de un punto y dos vectores (el vector normal se obtiene a partir del producto vectorial de los dos vectores).

-Crear a partir de un punto y un vector.

-Obtener alguna de sus componentes: obtenerPunto y obtenerVector.

-Distancia a un punto.

-Distancia a una recta.

-Distancia a otro plano.

Se trata de crear una aplicación que incluya las definiciones anteriores y a través de pequeños ejemplos comprobar que funcionan correctamente