

NOMBRE: Cristhian Calle
CDO160: 7024

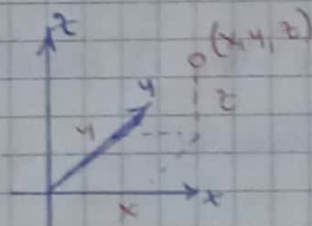
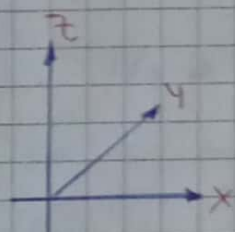
REPRESENTACIONES 3D

INTRODUCCION

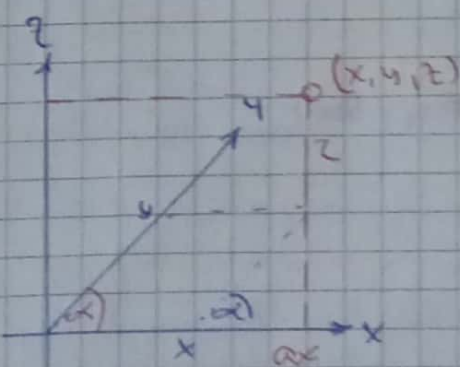
El modelado 3D es el proceso de desarrollo de una representación matemática de cualquier objeto tridimensional (ya sea inanimado o vivo) a través de un software especializado. Las escenas de gráficos pueden contener muchos clases diferentes de objetos, por ejemplo: flores, edificios, árboles, autos, aviones, agua, animales, etc. Para poder producir despliegues realistas de los objetos, es necesario utilizar representaciones que modelen con precisión dichos objetos.

Ejes

Las superficies cuadráticas y de polígonos proporcionan descripciones exactas para los objetos euclidianos sencillos como poliedros y elipsoides. Las superficies de spline y las técnicas de construcción son útiles para diseñar las alas de los aviones, engranajes y otras estructuras de ingeniería. De tal manera que se tiene en la siguiente figura lo que seña la representación en 3D.



Así, tal cual se indica de una manera más detallada en como es su demostración matemática, según sus formulas y mas especificas con sus atributos graficos



$$ax = x + y \cos(\alpha)$$

$$ay = y \sin(\alpha) + z$$

$$ax = x + py \cos(\alpha)$$

$$ay = py \sin(\alpha) + z$$

Procesos

Dando paso a una interpretación algorítmica para poder efectuar de una mge. ilustración gráfica, así, también para ejercer una practica de laboratorio se tiene el siguiente proceso.

Proceso Axonometria (float x, float y, float z, out float ax, out float ay)

```
}  
    ax = x*(0.6)*y + cos(0.8);  
    ay = (0.6)*y + sin(0.8)*z;
```

```
}
```

Creando así una clase que puede ayudar a efectuar de una mejor manera, y serviría como apoyo para las próximas ilustraciones que generarían en el transcurso de las prácticas de laboratorio.

Creando así la clase heredando de la clase padre Vector.

class vector3D: Vector

```
}
```

z0 ∈ ℝ

Proceso Axonometria.

Proceso Encender.

```
}
```

De tal manera que efectuara simultáneamente los procesos integrados en la clase Vector3D

Siendo parte una de ellos el proceso vector3D. Encender

Proceso vector3D. Encender

```
}
```

float ax;

float ay;

int Sx;

int Sy;

Axonometria (x0, y0, z0, out ax, out ay);

Pantalla (ax, ay, out Sx, out Sy);

if (Sx == 0 && Sx < 600 && Sy == 0 && Sy < 460)

```
{
```

EncenderPixel(Sx, Sy, color0);

```
}
```

```
}
```


SEGMENTOS 3D

Un segmento es la parte de una curva de spline que se encuentra entre dos de sus vértices. En el nivel de spline editable (segmento), se puede seleccionar uno o varios segmentos y moverlos, rotarlos y escalarlos mediante métodos estándar.

Teniendo así la siguiente representación de una clase que ayudara a graficar segmentos en 3D. Para seguir entendiendo se detalla de la siguiente manera:

Clase Segmento 3D

```
class segmento3D: Vector3D
```

```
{
```

```
    x f  $\in \mathbb{R}$ 
```

```
    y f  $\in \mathbb{R}$ 
```

```
    z f  $\in \mathbb{R}$ 
```

```
    Proceso Encender
```

```
}
```

Obteniendo esta clase, nos percatamos que toma como clase padre a la clase Vector3D la cual anteriormente se declaró. Por lo que se tiene en consideración ex que sus variables internas pertenecen a los números reales.

Para tener un poco más de guía se interpretan los procesos indicados los cuales nos permitirán generar la vista al momento de generar la práctica de laboratorio.

Proceso Segmento. Encender.

```
Proceso Segmento. Encender
```

```
{
```

```
    float t = 0;
```

```
    float dt = 0.001;
```

```
    Vector v3d = new Vector3D(0, 0, 0, color 0);
```

```
do {
```

```
    v3d.x0 = x0 * (1-t) + x1 * t;
```

```
    v3d.y0 = y0 * (1-t) + y1 * t;
```

```
    v3d.z0 = z0 * (1-t) + z1 * t;
```

```
    v3d.Encender;
```

```
    t = t + dt;
```

```
} while (t < 1)
```

```
}
```