

Práctica 4: Interpolación

Introducción

El objetivo de esta práctica es obtener una primera impresión de las posibilidades de la interpolación para la animación.

Notas Generales

Esta práctica está diseñada para ser realizada en clase y no va a ser corregida o calificada. Sin embargo debes intentar completar todos los puntos y aprovechar el tiempo en el aula. En caso contrario será motivo para suspender la asignatura.

Interpolación lineal: Curvas

En esta práctica vas a continuar con el proyecto de transformaciones. Recupera el proyecto y si no lo habías terminado consulta con el profesor.

1. Añade la siguiente estructura (array de puntos)

```
typedef struct {  
    std::vector<VECTOR3D> P;  
} CURVE;
```

2. Escribe la siguiente función utilizando las notas de clase para curvas con 2, 3 y cuatro puntos, correspondiente a curvas Bezier rectas, parabólicas y Cúbicas:

```
VECTOR3D interpolate(CURVE curve, float t)
```

Puedes programar la función anterior mediante el algoritmo de Casteljau o mediante la expresión de Bezier.

3. Escribe la siguiente función:

```
void drawCurve(CURVE curve)
```

Utiliza la función **interpolate** para ello.

4. Crea curva de dos, tres y cuatro puntos y dibújalas para probar las funciones anteriores.
5. Crea un punto (rojo) que avance por la curva anterior. Utiliza la tecla '+' y '-' para hacer avanzar o retroceder el punto.