

**Informática Gráfica**  
**Ingeniería en Informática**  
Curso 09-10. Práctica 1

**Carácter:** Obligatorio.

**Fecha de entrega:** Miércoles 28 de Octubre.

**Objetivo:** glSkeleton2D, OpenGL básico, exploración de la escena, tiling.

**Descripción:** Una *hipotrocoide* es la curva que traza un punto situado a una distancia  $c$  del centro de un círculo móvil de radio  $b$  que rueda sin resbalar dentro de un círculo más grande y fijo de radio  $a$ . Se entiende que  $a > b$ . Sus ecuaciones paramétricas vienen definidas por las siguientes expresiones:

$$x = (a - b) \cdot \cos(t) + c \cdot \cos\left(t \cdot \frac{a - b}{b}\right)$$
$$y = (a - b) \cdot \sin(t) + c \cdot \sin\left(t \cdot \frac{a - b}{b}\right)$$

Nuestro objetivo es dibujar una de estas curvas, uniendo los puntos que resultan al evaluar sus ecuaciones sobre una muestra de valores del parámetro  $t$ . Concretamente, para dar una *vuelta*, dividiremos una circunferencia en un número de partes iguales  $n$ , para formar el conjunto de parámetros  $\{i \cdot 2\pi/n, 0 \leq i \leq n\}$ . De esta forma la figura surge al unir los puntos correspondientes a dos parámetros consecutivos. Desafortunadamente, este conjunto puede ser insuficiente para cerrar la figura por completo. De hecho puede demostrarse que, si  $a$  y  $b$  son números naturales, el mínimo número de vueltas que debemos completar es:

$$\frac{b}{\text{mcd}(a, b)}$$

Por ello, ampliaremos la muestra de parámetros para que cubra uniformemente todas las vueltas necesarias.

**Detalles de la implementación:** Tu solución debe comenzar calculando el número de vueltas que deben completarse para cerrar la figura correspondiente a los parámetros  $a$ ,  $b$  y  $c$  que elijas. Además, debe ofrecer un menú con las siguientes operaciones:

1. Aumentar la precisión de la figura. Esto se consigue ampliando el tamaño de la muestra de parámetros, o lo que es equivalente, aumentando  $n$ .
2. Acercarnos a la escena manteniendo el centro del área visible de la escena.
3. Alejarnos de la escena manteniendo el centro del área visible de la escena.
4. Desplazar el centro del área visible de la escena sin modificar su tamaño.
5. Centrar el área visible de la escena en el origen sin modificar su tamaño.
6. Embaldosar la ventana con el área visible de la escena actual. El usuario fijará el número de columnas de forma interactiva. Cada una de las baldosas debe respetar el radio del área visible de la escena que se usa como modelo.

Salvo durante el embaldosado, siempre debe dibujarse el centro del área visible de la escena.

Por último, tu implementación debe soportar los cambios que el usuario provoque en el tamaño del puerto de vista. Los requisitos son los siguientes:

1. El área visible de la escena no puede deformarse.
2. El centro del área visible de la escena debe mantenerse.

**Parte opcional:**

1. [+] Consigue que el color de los segmentos que componen la curva vaya cambiando progresivamente; y permite configurar los valores de los valores a, b y c en tiempo de ejecución.
2. [++] Añade un reloj para conseguir el efecto de animación, es decir, la curva se irá pintando poco a poco.

