

Informática Gráfica
Ingeniería en Informática
Curso 09-10. Práctica 2.2

Carácter: Obligatorio

Fecha de entrega: miércoles 2 de diciembre

Objetivo: gráficos interactivos, estructura de la escena, dibujo relativo, recorte

Descripción: Se trata de construir un editor de escenas que permita de forma interactiva:

1. Explorar la escena sin más que modificar su área visible mediante traslaciones y escalaciones. Como resulta habitual, las primeras mueven el centro del área visible sin modificar su tamaño actual y las segundas, al revés, varían el tamaño de la superficie visible sin variar su centro.
2. Añadir las siguientes figuras a la escena:
 - a. Poli-líneas especificadas por los puntos que el usuario vaya eligiendo con el ratón sobre el puerto de vista.
 - b. Polígonos regulares determinados por el número de lados, longitud del lado y el centro que el usuario fije con el ratón.
 - c. Espirales dadas por el tamaño del lado inicial, el incremento sucesivo en la longitud de los lados, el ángulo de giro, el número de iteraciones y el punto de partida establecido con el ratón.
 - d. Curvas hipotrocoides especificadas por los parámetros centro, a , b , c y n , que determinan respectivamente la posición de su centro, el radio de su círculo fijo, el radio de su círculo móvil, su offset, y el número de puntos equiespaciados que se toman en el rango $[0, 2\pi]$. Se entiende que $a > b$.
3. Editar una hipotrocoide cuando la distancia entre el punto que elige el usuario con el ratón y el centro de la curva es menor que $a + c$. La edición corresponde a una de las siguientes opciones:
 - a. Aumentar o disminuir la calidad de la curva, mediante la modificación del número de puntos equiespaciados que se toman en el rango $[0, 2\pi]$.
 - b. Girar la curva un ángulo determinado sobre su centro.
4. Editar una figura cuando la distancia entre alguno de sus vértices y el punto que elige el usuario con el ratón sea menor que una cantidad previamente fijada. La edición corresponde a una de las siguientes opciones:
 - a. Seleccionar la figura, que aparecerá con otro color
 - b. Borrar la figura seleccionada.
5. Recortar la escena eliminando todo lo que quede fuera de un rectángulo fijado con el ratón. Como caso excepcional, las hipotrocoides recuperarán todos los puntos que perdieron tras un recorte, cuando se solita aumentar o disminuir su calidad.
6. Archivar la escena para poder recuperarla posteriormente. Como es habitual, el menú de Ficheros deberá permitir abrir, guardar, empezar una escena nueva y salir. En el caso de haber modificado la escena actual, el usuario recibirá el aviso correspondiente si solicita empezar o abrir otra escena.

Detalles de la implementación: Estructura de la información

Debes estructurar la escena adecuadamente otorgando a cada clase el comportamiento y los atributos que le correspondan. Entre otras debes usar las siguientes clases:

1. *Escena*, que contenga las figuras que incluye repartidas en dos listas: *ldls* para las poli-líneas, los polígonos regulares y las espirales, y *lhs* para las hipotrocoides. Los elementos de *ldls* serán dibujos de líneas, mientras que las de *lhs* serán hipotrocoides. Cada lista tendrá un color característico que nos servirá para distinguir las figuras de ambas listas. Otros atributos de la escena serán los toques que determinan su área visible, su nombre y un booleano que indique si ha sido modificada.
2. *Dibujo de líneas*, que contenga una lista con todos los segmentos que contiene.
3. *Hipotrocoide*, que almacene los parámetros que la definen (centro, a, b, c y n) y la lista de los segmentos que la componen. Observa que este último atributo indica que una hipotrocoide es un dibujo de líneas especial, por lo que debes implementar esta clase heredando de *Dibujo de líneas*.
4. *Segmento*, que guarde los puntos que determinan sus extremos.
5. *PV*, que guarde las coordenadas de un punto o de un vector en dos dimensiones.

Otras clases auxiliares interesantes son:

1. *Lista*, que permita gestionar listas polimórficas que sirvan a la vez para segmentos o dibujos de líneas. Puedes usar las listas predefinidas en la API de C++ o implementar las tuyas propias mediante plantillas (*templates*).
2. *Lápiz*, cuyos atributos de posición y dirección habitual permitan realizar dibujos relativos.

Detalles de la implementación: Diseño algorítmico

1. La interacción con el gráfico ha de tener en cuenta la diferencia entre el puerto de vista y el área visible de la escena. Por ello, los puntos seleccionados con el ratón deben transformarse adecuadamente en puntos de la escena. Esto debe seguir funcionando tras originarse un evento *OnResize* o tras aplicar una traslación o un zoom.
2. El método de la clase PV que sirve para rotar el punto que recibe el mensaje es el siguiente (x e y son las coordenadas del punto, centroRot es el centro de rotación, y ang es el ángulo):

```
void PV::rotaP(PV* centroRot, float ang){
    float cx= centroRot->getX();
    float cy= centroRot->getY();
    float dx=(1.0-cos(ang))*cx + sin(ang)*cy;
    float dy=-sin(ang)*cx + (1.0-cos(ang))*cy;
    float x1= x*cos(ang)-y*sin(ang)+dx;
    float y1= x*sin(ang)+y*cos(ang)+dy;
    x=x1;
    y=y1;
}
```

3. El recorte debe basarse en el algoritmo de Cohen-Sutherland.
4. Puedes usar objetos de las clases predefinidas *TOpenDialog* y *TSaveDialog* para implementar las herramientas correspondientes a la gestión de ficheros.

Detalles de la implementación: Etapas de desarrollo

Lo que sigue son las etapas que puedes seguir para desarrollar la implementación cómodamente. Como es habitual resulta conveniente probar el código correspondiente a una etapa antes de pasar a la siguiente fase.

- Etapa I: estructura de la escena (clases principales, más la clase *Lista*).
- Etapa II: exploración de la escena (escalación y traslación).
- Etapa III: dibujo de poli-líneas, incluyendo transformación desde el puerto de vista a la escena.
- Etapa IV: inserción de espirales.
- Etapa V: inserción de hipotrocoides.
- Etapa VI: edición de hipotrocoides (modificar su calidad y girarla).
- Etapa VII: edición de figuras (seleccionar y borrar figuras).
- Etapa VIII: recorte de figuras.
- Etapa IX: gestión de ficheros (nuevo, abrir, guardar y salir).

Parte opcional:

1. [+] Implementa la construcción de los *poli-arcos* explicados en clase.
2. [+] Implementa un nuevo zoom capaz de modificar el área visible de la escena y permitir que aparezcan por completo todas las figuras. Como es habitual en las escalaciones, no se permiten deformaciones, ni cambiar el centro del área visible de la escena.
3. [+] Añade al menú de edición de figuras los siguientes comandos: copiar, pegar y mover la figura seleccionada.
4. [+] Si ya has resuelto el apartado anterior, admite la posibilidad de deshacer la última edición (incluyendo la de borrado).