

Informática Gráfica
Ingeniería en Informática
Curso 08-09. Práctica 2

Carácter: Obligatorio

Fecha de entrega: Miércoles 26 de Noviembre (parte obligatoria)
Miércoles 3 de Diciembre (parte opcional)

Objetivo: gráficos interactivos, estructura de la escena, diseño 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 dibujos de líneas a la escena entre los que se incluyan:
 - 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, y
 - 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.
3. Recortar la escena eliminando todo lo que quede fuera de un rectángulo fijado con el ratón.
4. Editar la escena incluyendo herramientas para:
 - a. seleccionar un dibujo de líneas al pulsar el botón izquierdo del ratón sobre uno de sus vértices, y
 - b. borrar el dibujo de líneas seleccionado.
5. Transformar el dibujo de líneas seleccionado mediante la aplicación de un patrón sobre cada uno de los segmentos que lo componen. Los patrones requeridos son fractales clásicos que se muestran en la Figura 1. En la columna de la derecha aparece el dibujo de líneas en que se convierte cada uno de los segmentos que aparecen a la izquierda. De arriba a abajo los patrones se llaman Koch1, Koch2 y Dragón.

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 una lista de los dibujos de líneas que incluye. Los toques que determinan el área visible deben ser atributos de la escena.
2. *Dibujo de líneas*, que contenga una lista de todos los segmentos que contiene.
3. *Segmento*, que guarde los puntos que determinan sus extremos.
4. *Punto*, que guarde las coordenadas de un punto en dos dimensiones.

Otras clases auxiliares serán:

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 actual permitan realizar figuras mediante diseño relativo.

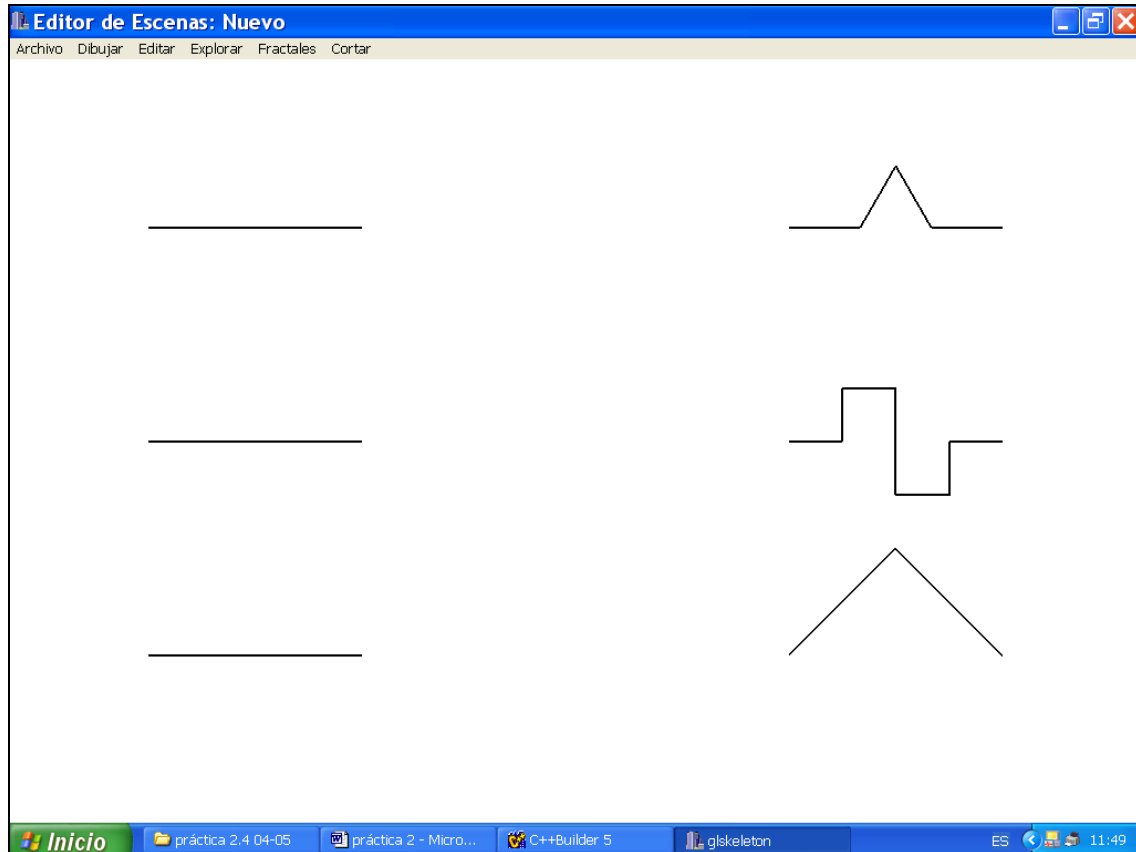


Figura 1

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. Para dibujar poli-líneas, polígonos y espirales puedes usar un objeto de la clase *Lápiz*.
3. El recorte debe basarse en el algoritmo de Cohen-Sutherland.
4. La selección de un dibujo de líneas debe realizarse por aproximación y no por coincidencia total; es decir, para que un dibujo de líneas sea elegido basta con que el punto ubicado por el usuario esté muy próximo a uno de sus vértices.
5. Las transformaciones correspondientes a los patrones de la Figura 1 requieren el uso de un objeto de la clase *Lápiz*. Observa que si l es el segmento que está siendo transformado, el lápiz debe empezar posicionándose sobre el origen de l y orientándose en la dirección de l .

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: exploración de la escena (escalación y traslación)
- Etapa II: estructura de la escena (clases principales, más la clase *Lista*)
- Etapa III: dibujo de poli-líneas, polígonos y espirales, incluyendo la clase *Lápiz* y la transformación desde el puerto de vista a la escena
- Etapa IV: recorte de líneas
- Etapa V: herramientas de edición (seleccionar y borrar)
- Etapa VI: fractales (Koch1, Koch2 y Dragón)

Parte opcional:

1. [+] Añade un botón para imprimir por impresora, o en un archivo, el área visible de la escena actual.
2. [+] Añade la posibilidad de seleccionar un polígono regular pulsando con el ratón cualquier punto de su interior. Esto supone diferenciar los polígonos regulares de las poli-líneas, ya que sólo los primeros permiten esta operación. Observa que tras ejecutar el algoritmo de recorte algunos polígonos regulares pueden dejar de serlo, incluso pueden ni considerarse regiones rellenas. Algo parecido sucede al aplicar los patrones de la Figura 1, ya que los polígonos dejan de ser regulares. Piensa cómo solucionar estos problemas.