

Caracas, 24 de Junio de 2022.

Proyecto # 1

Se requiere que usted realice una aplicación con las siguientes características:

- Se debe permitir la creación de las siguientes primitivas 2D: líneas, círculos, elipses, rectángulo, triángulo y curvas de Bézier (usar los número 1..6 + interfaz). Una vez creada la primitiva su estado es “seleccionada” por defecto. Se debe poder configurar el color de fondo del lienzo (usar B mayúscula para luego introducir el color, y/o utilizar un color picker RGB en la interfaz gráfica). Igualmente se debe poder cambiar el color de línea y color de relleno de las primitivas, de manera tal que la nueva primitiva creada (o la primitiva seleccionada) tome ese color. Usar las teclas ‘c’ y ‘f’ para cambiar el color de borde (c) y relleno (f), además de color picker in la interfaz gráfica.
- Las curvas de Bézier se deben desplegar utilizando el algoritmo de de Casteljau. El resto de las primitivas deben desplegarse de forma incremental utilizando aritmética entera.
- El proyecto soporta 2 modalidades: hardware/software. Si la opción hardware está habilitada, las primitivas gráficas pueden desplegarse usando glBegin(GL_LINES), glBegin(GL_QUADS), etc. Como excepción, CCircle y CEllipse siempre se desplegarán por software, pixel a pixel. Si la opción software está habilitada, solo las funciones “putPixel” y “setColor” de la clase CShape pueden ser utilizadas, lo que quiere decir que todas las primitivas serán desplegadas pixel a pixel. Usar la tecla ‘h’ para habilitar/deshabilitar hardware + interfaz.
- Cada primitiva puede ser seleccionada con el ratón en cualquier momento. Cuando este evento ocurre se debe permitir:
 - a. Modificar la primitiva: Tamaño (moviendo los puntos de control), posición (trasladando la figura, evento mouseMove – dragging-), color de borde, color de relleno (sólo para el caso de elipses, círculos o triángulos).
 - b. Desplegar indicadores de que la primitiva está seleccionada (marcas en los vértices de la línea, en los vértices del bounding box de la elipse, del triángulo o en los puntos de control de Bézier).
 - c. Enviar al frente o al fondo la primitiva, ya sea vía menú, o con shortcuts (se sugiere ‘f’ para traer al frente, y ‘b’ para enviar al fondo, y a la vez un par de botones en la interfaz).

- d. Mover la primitiva “un nivel” hacia adelante o hacia atrás en la lista de despliegue (se sugiere ‘+’ para traer al frente, y ‘-’ para enviar al fondo, y a la vez un par de botones en la interfaz).
 - e. Deseleccionar la primitiva (haciendo clic en otra primitiva, o en el fondo, o presionando la tecla ‘u’).
 - f. Eliminar la primitiva (se sugiere la tecla *delete* o suprimir, y vía botón de menú).
- La escena 2D puede ser guardada y cargada posteriormente (teclas ‘s’, ‘L’ + interfaz). Favor utilizar archivos de texto, de manera que podamos crear una escena editando el archivo de texto, y luego cargarla en la aplicación.
 - Mediante una opción, se pueden eliminar todas las primitivas desplegadas (tecla ‘x’ y botón de “clear”).
 - Cambiar el puntero del ratón, según la acción a realizar (“mouse over”, trasladar primitiva, mover vértice o punto de control, ...).

Entrega:

- Se debe entregar (18 de Julio) el código fuente en un archivo zip, y el ejecutable, así como todo lo necesario para su compilación y ejecución. Usar PortalASIG para enviar este archivo ZIP. Este no debe ocupar más de 5MB.
- En todo caso y conociendo los problemas que hay con la plataforma, como backup, enviar LINK de dropbox del mencionado ZIP a mi correo recs34@gmail.com (únicamente el LINK de dropbox), para que no rebote el correo. Indicar cédula y nombre en dicho correo, colocando como subject “Proyecto 1 – Gráfica”.
- Se penaliza 1 punto por día de retardo en la entrega.

Criterio de evaluación:

I. RENDERING (10 puntos)

- Rendering de primitivas correctamente por software (9 puntos)
 - Línea: 1.5 puntos
 - Círculo: 0.5 puntos
 - Elipse: 1.5 puntos
 - Triángulo: 2.5 puntos

- Rectángulo: 0.5 puntos
- Curva: 2.5 puntos
- Rendering de primitivas por hardware (1 punto)
 - Solo línea (GL_LINES), triángulo(GL_TRIANGLES), rectángulo (GL_QUADS), Bézier (GL_LINE_STRIP). Círculo y Elipse se desplegarán únicamente por “software”.

II. INTERFAZ (8 puntos)

- Introducción de los objetos con el ratón y modificar sus puntos de control (3 puntos)
- Mostrar correctamente cuál objeto está seleccionado, y deseleccionar (1 punto)
- Enviar al frente, atrás, adelantar, retroceder (1 punto)
- Salvar/Guardar archivo correctamente (2 puntos)
- Cambiar el puntero del ratón como una aplicación pro (1 punto)

III. CALIDAD DEL SOFTWARE (2 puntos)

- Uso correcto de clases abstractas, métodos virtuales, apuntadores, destructores, comentarios en código, legibilidad, eficiencia, robustez (sobre todo en cuando a la lectura/escritura del archivo), diseño, validaciones, adaptabilidad, etc.

MISC:

1. Los aspectos de interfaz están condicionados a las clases implementadas. Si se implementa únicamente la Figura “CLine”, esto tendrá un peso de 1/6, pues son 6 figuras las que se debieron implementar. Por ejemplo, Introducción de los objetos con el ratón estará evaluado sobre 2/6 puntos y no sobre 2. Lo mismo aplica a la calidad del software. Se recomienda usar AntTweakBar para el manejo de botones y controles de interfaz: <http://anttweakbar.sourceforge.net/>
2. Note que el proyecto podría ser realizado con o sin AntTweakBar, pues a través del teclado y mouse se pueden realizar todas las funcionalidades. Solo un punto sobre la nota de interfaz será penalizado por no utilizar AntTweakBar.
3. Pensando siempre en que ese código lo podría “mejorar”, “actualizar” otra persona en el mundo, mantener el código, menú, comentarios, todo en inglés, incluso las “teclas” con su funcionalidad en inglés (e.g. L = “load”).
4. Estoy a su disposición para ayudarlos en cualquier momento si se llegan a trabar. Pueden usar preferiblemente PortalAsig o Discoord para preguntar. Así todos podemos aprender de todos.
5. Para navegar en la estructura de directorio (para LOAD/SAVE), recomiendo utilizar tiny file dialog: <https://sourceforge.net/projects/tinyfiledialogs/>. Pero pueden intentar con otro.

6. Formato de archivo por línea de texto:

<primitiva> <puntosXY> <colorRGB> [color_rellenoRGB]

“color_rellenoRGB” solo aplica a rectángulo, elipse, triángulo y círculo.
Se usará adicionalmente un “tag” para especificar el color de background.

Ejemplo:

```
BACKGROUND 0.0 0.0 0.0
LINE 80 20 43 322 1.0 0.5 0.2
BEZIER4 0 3 0 40 3 23 200 100 1.0 0.0 0.0
RECTANGLE 0 0 100 200 1.0 0.5 0.2
FILLED_RECTANGLE 90 100 10 10 0.0 1.0 0.0 0.0 0.0 0.6
TRIANGLE 240 200 200 240 280 240 1.0 0.5 0.2
FILLED_TRIANGLE 100 100 120 130 80 70 1.0 0.5 0.2 0.0 1.0 0.0
ELLIPSE 100 100 120 110 0.0 0.0 0.0
FILLED_ELLIPSE 300 300 400 350 1.0 0.0 0.0 0.0 1.0 0.5
CIRCLE 50 60 60 70 1.0 1.0 1.0
FILLED_CIRCLE 60 70 80 90 1.0 1.0 1.0 0.0 0.0 0.0
```

Note que los puntos son enteros positivos de la forma x y x y ... mientras que los colores son 3 números en punto flotante entre 0 y 1. En el caso de BEZIER, el número de puntos debe variar entre 2 y 20. Así que el número de puntos es especificado en el nombre del objeto (BEZIER4 si son 4 puntos, por ejemplo). Debe haber al menos un espacio entre cada par de elementos (palabra o número) en el archivo. En este proyecto utilizaremos rectángulos alineado a los ejes, así que es suficiente especificar 2 esquinas opuestas del mismo. Favor respetar el formato del archivo, pues se harán pruebas con archivos predefinidos (como el del ejemplo). Cualquier cambio en el formato será penalizado con 5 puntos. De haber un error en alguna línea en el archivo (por ejemplo, una coordenada negativa, ausencia de alguna componente, color inválido, primitiva inválida, etc.) se ignora esa línea de texto y se indica el error por consola.

ÉXITOS

R.C./rc