

# Proyecto 1: Manejo de Polígonos en 2 Dimensiones

En este proyecto, Ud. deberá desarrollar y programar algoritmos para el despliegue de polígonos en dos dimensiones. Esto incluye relleno de polígonos con colores y con texturas. Toda la programación debe realizarse en C sobre Linux.

## I. EL MAPA

Este programa desplegará un mapa de Costa Rica con la división política en provincias. Cada provincia se representará por uno o más polígonos. A su vez, cada polígono se implementará como una secuencia de vértices en coordenadas universales (para un despliegue de calidad se estima que debería haber más de mil vértices en total). Inicialmente, se establecerá una ventana en coordenadas universales lo suficientemente amplia como para que todo el mapa aparezca en el despliegue. Las dimensiones de la ventana y del *framebuffer* deben ser de la misma proporción horizontal-vertical para que no haya distorsiones de la imagen.

## II. LAS PROVINCIAS

El mapa podrá ser desplegado en 3 modalidades diferentes:

- **Sin colorear:** en este, caso sólo se desplegarán los bordes que forman cada polígono (utilizando Bresenham).
- **Coloreado:** a cada provincia se le asigna un color diferente en el cual se rellenarán.
- **Texturas:** cada provincia se rellenará tomando “texeles” de un archivo gráfico diferente para cada una.

## III. INTERACCIÓN CON EL USUARIO

Con alguna interfaz sencilla (investigar las posibilidades de lectura de teclado y mouse de MESA/GLUT), el usuario podrá interactuar con el programa. Las operaciones posibles son:

- **Cambio de modo:** el usuario indicará si desea que el despliegue sea hecho sin colorear, coloreado o con texturas.
- **Zoom in/Zoom out:** la ventana establecida podrá ser ampliada (*zoom out*) o reducida (*zoom in*). El mapa será redibujado.
- **Pan Left/Right/Up/Down:** la ventana establecida podrá moverse hacia la izquierda, derecha, arriba o abajo.
- **Rotación:** el usuario podrá hacer rotar el mapa en el sentido de las manecillas del reloj o en sentido inverso.
- **Reiniciar:** volver a la vista inicial del mapa.
- **Terminar:** fin del programa.

El usuario podrá mezclar estas operaciones de la manera que desee. Por ejemplo, esto significa que se puede colorear el mapa o rellenar con texturas en cualquier instante, o hacerlo rotar coloreado o no, o ampliarlo con cualquier ángulo, etc.

**Trabajo extra opcional 1:** hacer que el Zoom in/Zoom out tenga poco incremento o decremento de la ventana cada vez que se pida la operación (modo lento) o mediante alguna interfaz apropiada, por ejemplo presionando *shift* al mismo tiempo, hacer que este incremento o decremento sea más marcado (modo rápido).

**Trabajo extra opcional 2:** hacer que el Pan en cualquier sentido tenga poco incremento o decremento de la posición de la ventana cada vez que se pida la operación (modo lento) o mediante alguna interfaz apropiada, por ejemplo presionando *shift* al mismo tiempo, hacer que este incremento o decremento sea más marcado (modo rápido).

**Trabajo extra opcional 3:** hacer que la rotación del mapa en cualquier sentido sea por pocos radianes cada vez que se pida la operación (modo lento) o mediante alguna interfaz apropiada, por ejemplo

---

presionando *shift* al mismo tiempo, hacer que la rotación sea por una mayor cantidad de radianes (modo rápido).

#### IV. REQUISITOS INDISPENSABLES

La ausencia de uno solo de los siguientes requisitos vuelve al proyecto “no revisable” y recibe un 0 de calificación inmediata:

- Todo el código debe estar escrito en C
- El proyecto debe compilar y ejecutar en Linux
- Todos los algoritmos gráficos deben haber sido desarrollados por los estudiantes. El uso de alguna biblioteca (MESA / GLUT o equivalente) se limita a crear una ventana, alterar el color de los píxeles, encender el píxel y manejar la interacción con el usuario.
- No debe dar “Segmentation Fault” bajo ninguna circunstancia.

**Trabajo extra opcional 4:** hacer la demostración en una máquina que levante Linux de manera real (puede ser dual), es decir no usar máquinas virtuales.

#### V. FECHA DE ENTREGA:

Demostraciones en clase el **Miércoles 7 de setiembre**.

Organice su proyecto en un directorio cuyo nombre sean los dos apellidos de los miembros del grupo (e.g. `torres-rojas`). Usando el comando `tar`, cree un `.tgz` de este directorio. Cuando este archivo sea expandido se creará exactamente el mismo directorio.

Enviar este `.tgz` por e-mail a `torresrojas.cursos@gmail.com` antes de la hora de inicio de las demostraciones. Ponga como *subject*: “COMPUTER GRAPHICS - Proyecto 1 - ”< *Fulano* > – < *Mengano* >” donde *Fulano* y *Mengano* son los 2 miembros del grupo.