



Práctica 1.- Introducción a GLUT y Rasterización de Primitivas 2D.

Objetivos:

Introducir al alumno/a en el entorno de desarrollo gráfico GLUT/OpenGL y programación de primitivas básicas 2D: trazado de rectas (DDA, Punto Medio, Bresenham ...).

Requerimientos:

Conocimientos de programación en C.

Conocimientos básicos de la GLUT así como de su filosofía de desarrollo.

Algoritmos de renderizado de rectas: DDA, Punto-Medio, Bresenham.

Realización:

Actividad 1.

Realizar una aplicación gráfica capaz de visualizar una malla uniforme (rejilla [grid] 2D) con distinta densidad de mallado, para una resolución fija.. Para ello se deberá partir del esquema proporcionado (`base.c`) e implementar distintas funciones para el dibujo de líneas basado en el algoritmo de Punto Medio-Bresenham. Deberás implementar las versiones para $m < 1$ y para $m > 1$

El prototipo de la función de trazado de rectas a implementar será el común, asignando el color negro al trazado:

```
draw_Bresenham_line_menor1(int x0, int y0, int x1, int y1);  
draw_Bresenham_line_mayor1(int x0, int y0, int x1, int y1);
```

Una vez realizadas ésta, necesitaremos incluir la rutina de dibujo del grid:

```
draw_grid(int gap_x, int gap_y, float ang); /*Dibuja un grid de  
espaciado (x,y) pixels y ángulo (ang) radianes*/
```

El programa deberá ser sensible a los siguientes eventos:

Ratón

botón izquierdo: aumenta en 5 pixels el espaciado del grid (tanto en x como en y).

botón derecho: reduce en 5 pixels el espaciado del grid (tanto en x como en y).

Teclado

tecla '+' aumenta en $\pi/36$ radianes la pendiente de las rectas del grid.

tecla '-' reduce en $\pi/36$ radianes la pendiente de las rectas del grid.

tecla 'e' termina el programa.

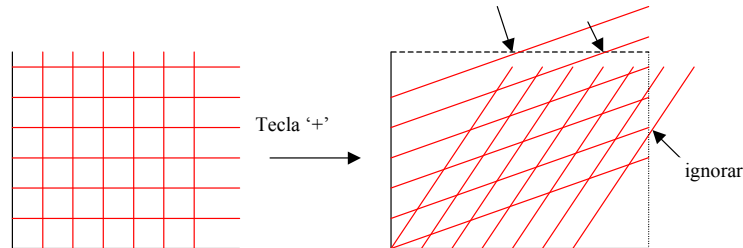


Notas Importantes:

1º Nos restringimos a dibujar rectas en el primer cuadrante. Así cuando pulsemos la tecla '+', la pendiente de las rectas horizontales se incrementan y las verticales se decremantan. Así evitaremos calcular Bresenham para pendientes negativas.

2º No es necesario tener en cuenta en esta práctica el hecho de que las rectas se recortarán en las aristas superior e inferior de la ventana.

Igualmente no es necesario rellenar con nuevas rectas el hueco que se va abriendo alrededor de la bisectriz del primer cuadrante, conforme se cambia de pendiente con las teclas.



Actividad 2 (OPCIONAL)

Dado un formato de fichero donde se encuentran descritos geometrías de polilíneas, realizar una rutina que dibuje las polilíneas en el área de dibujo.

Considera el archivo de datos geometricos *dino.dat* que contiene la especificación de polilíneas según el siguiente formato:

```
24      número de polilíneas del fichero
4       puntos de la polilínea
23 116 punto
56 120
45 78
67 145
3       puntos de la polilínea
345 234
238 321
126 210
```

...
Implementar una rutina para leer el fichero y utilizar el algoritmo de Bresenham en sus diferentes casos (pendientes positivas y negativas, mayores y menores que 1) para dibujar las rectas que forman cada polilínea.

Actividad 3 (OPCIONAL)

Modifica el programa de la Actividad 1 para que dibuje también círculos con la siguiente interacción:

- Al pulsar la tecla 'c' se borra la pantalla y se dibuja una circunferencia usando Bresenham de radio 15.
- Al pulsar la tecla 'r' se redibuja la circunferencia aumentando el radio el 10 unidades.
- Al pulsar la tecla 'd' se redibuja la circunferencia disminuyendo el radio en 10 unidades (comprobar que el radio no es negativo)
- Al pulsar otra vez la tecla 'c' volvemos al dibujado de la rejilla en el estado inicial.