



## Práctica 2.- Estudio de Algoritmos de relleno de polígonos en dispositivos de barrido

### Objetivos:

Estudiar el comportamiento de algunos algoritmos de relleno de polígonos y figuras cerradas.

### Requerimientos:

- Algoritmo del punto medio de generación de circunferencias
- Algoritmo de la semilla en alguna de sus versiones.

### Enunciado:

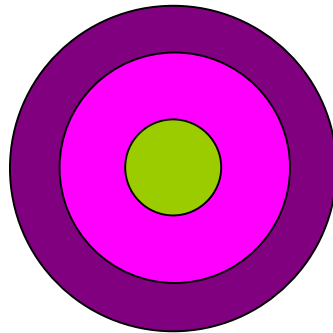
Vamos a dibujar en el puerto de vista de nuestra ventana una diana. Para ello consideraremos el algoritmo del punto medio para circunferencias, con el que generaremos las fronteras de cada una de las áreas de la diana. Considera tres áreas de diferente color

Area del centro: Color Rojo. Frontera a 50 pixels del centro

Segunda Area concéntrica: color Verde. Frontera a 100 pixels del centro

Tercera área concéntrica: Color Azul. Frontera a 150 pixels del centro

Frontera de cada área: Color Negro



Implementa una versión del algoritmo de la semilla para rellenar los colores de cada círculo concéntrico. La versión que mejor será considerada es la del uso de una pila para evitar la recursión.

### Dinámica del programa

Al iniciar el programa, se dibujarán sobre una ventana de fondo blanco las tres circunferencias que forman la diana sin colorear, utilizando el algoritmo del punto medio (Bresenham) para circunferencias.

Pinchando con el botón izquierdo del ratón en cada uno de los círculos, se adquiere la semilla inicial para pintar el área de un color. Si se pincha en un punto exterior o sobre un área coloreada, el sistema puede advertir sacando un mensaje por la ventana de mensajes de que la acción es incorrecta.

Como parte complementaria puedes modificar el programa para que las circunferencias sean generadas por la librería OpenGL y utilizar las funciones de relleno de área de que dispone dicha librería



**Notas importantes:**

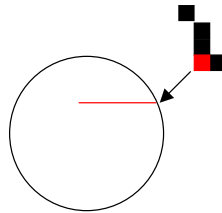
1ª La instrucción de lectura del color de píxel no funciona en las librerías de la openGl (al menos para el S.O. Windows).

Esta instrucción es

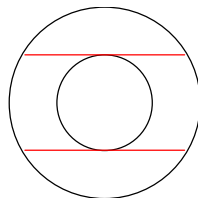
```
glReadPixels(x, y, 1, 1, GL_RGB, GL_FLOAT, rgb);
```

para desarrollar la práctica, declarar una matriz de tipo short de dimension TAMAÑO\_X x TAMAÑO\_Y donde se almacenará el estado de color de cada píxel (0 si tiene el color del fondo, 1 si tiene el color de frontera, 2 si ha sido pintado)

2ª Para desarrollar el algoritmo de relleno mejorado (usando una pila) hay que tener en cuenta que debido a la forma de las fronteras (circunferencias) no se debe consultar el estado del píxel inmediatamente superior e inferior al del borde actual, sino que por precaución se debe consultar el de algunos pixeles antes (por ejemplo 10 pixeles es aceptable)



3º Para evitar el efecto de “sombra” en algunas regiones es necesario coger semillas en las dos fronteras de cada tramo. Realmente solo es necesario en esta geometría el cogerla exactamente en la línea de la corona que está inmediatamente encima o debajo de la frontera de una circunferencia más interior.



**Parte Opcional:**

Desarrolla un código para diseñar polígonos a mano alzada y luego rellenarlos de un color cualquiera (no hace falta pedir la opción del color al diseñador).

Para ello confeccionar polígonos pinchando con el botón izquierdo del ratón en el área de la ventana para introducir un nuevo vértice y pinchar con el botón derecho para indicar el ultimo. Tras esto, se unirán con líneas rectas todos los vértices (usando Bresenham de la práctica 1).

Una vez confeccionado el polígono cerrado, bastará pinchar en el interior con el botón izquierdo del ratón tras pulsar la tecla ‘s’, para conseguir la semilla del algoritmo de coloreado.

Para borrar el polígono realizado puedes activar la tecla ‘b’. Tras pulsarla se envía una orden de redibujado de la pantalla.