

# Guía de Usuario – PU1

## **1. Introducción**

Este sistema permite graficar figuras geométricas en un plano cartesiano mediante distintos algoritmos:

- Líneas: Método Directo y DDA
  - Círculos: Incremental y Punto Medio
  - Elipses: Punto Medio
- 

## **2. Requisitos del Sistema**

- Sistema operativo: Windows, Linux o MacOS
  - Software: Code::Blocks o compilador compatible con C++
  - Conocimientos básicos de programación
- 

## **3. Instalación**

1. Descargar el repositorio desde GitHub.
  2. Abrir Code::Blocks y crear un proyecto nuevo en C++.
  3. Copiar los archivos de la carpeta src/ al proyecto.
  4. Compilar (Build) y ejecutar (Run) el programa.
- 

## **4. Funcionamiento**

### 4.1. Visualización

Al ejecutar el programa:

- Se abre una ventana con un plano cartesiano.
- Se dibujan automáticamente las figuras con coordenadas predeterminadas.
- Cada punto puede ser mostrado en consola si se utiliza la versión en PSeInt.

### 4.2. Parámetros Configurables

- Líneas:  $x_i$ ,  $y_i$ ,  $x_f$ ,  $y_f$
  - Círculos:  $x_c$ ,  $y_c$ ,  $r$
  - Elipses:  $x_c$ ,  $y_c$ ,  $r_x$ ,  $r_y$
  - Color:  $r$ ,  $g$ ,  $b$
  - Grosor de línea/píxel
-

## **5. Menú y Atajos**

Aunque el programa original se ejecuta automáticamente, se pueden agregar opciones interactivas como un menú para seleccionar algoritmos:

Opción	Descripción	Atajo
1	Mostrar/ocultar la cuadrícula	G
2	Mostrar/ocultar los ejes	E
3	Limpiar todas las figuras	C
4	Exportar la imagen en formato PPM	S

**Nota:** Para versiones futuras se puede implementar un menú gráfico con opciones seleccionables mediante teclas o botones.

---

## **6. Ejecución**

1. Seleccionar el algoritmo deseado mediante el menú o modificar los parámetros en el código.
  2. Observar los resultados en la ventana gráfica.
  3. Los puntos se redondean automáticamente para ubicar correctamente los píxeles.
  4. Se pueden repetir los procesos para distintas figuras sin reiniciar el programa.
- 

## **7. Consejos de Uso**

- Mantener la versión en PSeInt para simular y comprobar los cálculos paso a paso.
- Para personalizar colores, modificar los valores de r, g y b en el código.
- Ajustar el grosor de línea si se desea que los trazos sean más visibles.