



**UNIVERSIDAD NACIONAL DEL ALTIPLANO**  
**FACULTAD DE INGENIERIA ESTADISTICA E INFORMATICA**



**GRAFICADOR ASCII**

**Autores:**

**Aquino Sandoval Jean Carlos (214689)**

**Semestre: IV grupo: A**

**Docente: Ing. Torres Cruz Fred**

**Area: Programacion Numerica**

**Puno – Perú**

**2025**

# Graficador ASCII

## 1. Objetivo

Desarrollar una herramienta interactiva que permita al usuario ingresar dos funciones lineales o polinómicas y representarlas gráficamente en formato ASCII dentro de la misma aplicación. Además, resalta las intersecciones y los ejes de coordenadas, brindando una visualización rápida y clara de la relación entre las funciones.

## 2. Descripción general del código

El programa está implementado en Python usando Tkinter para la interfaz gráfica. Permite:

1. Ingresar dos funciones matemáticas en cajas de texto.
2. Analizar y preparar las expresiones para que puedan ser evaluadas correctamente en Python:
  - Conversión de potencias ( $\wedge \rightarrow **$ ).
  - Transformación de multiplicaciones implícitas ( $2x \rightarrow 2*x$ ).
  - Casos especiales como  $-x$  o  $+x$ .
3. Generar una representación gráfica ASCII en un área de resultados:
  - $*$  para la primera función.
  - $o$  para la segunda función.
  - $\#$  para la intersección de ambas funciones.
  - $|$  para el eje Y,  $-$  para el eje X y  $+$  para el origen.
4. Limpiar entradas y resultados mediante un botón.

## 3. Estructura y secciones del código

### 3.1 Función `preparar_expresion(expr: str) -> str`

- Recibe una cadena con una función matemática y la transforma para evaluarse correctamente en Python.
- Pasos:
  - Elimina espacios.
  - Convierte  $\wedge$  a  $**$ .
  - Convierte  $-x$  y  $+x$  a  $-1*x$  y  $+1*x$ .
  - Añade multiplicación implícita ( $x \rightarrow *x$ ).

- Corrige casos donde `**x` se pueda duplicar.
- Garantiza que expresiones como  $3x^2 + 2x - 5$  sean evaluables en Python.


### 3.2 Función graficar()

- Obtiene las funciones de las entradas (entrada1 y entrada2).
- Valida que ambas funciones no estén vacías; si lo están, muestra un warning.
- Convierte las expresiones usando `preparar_expresion`.
- Define un rango de coordenadas `xmin` a `xmax` y `ymin` a `ymax`.
- Recorre todos los valores de `y` de arriba hacia abajo y de `x` de izquierda a derecha:
  - Evalúa `y1` y `y2` para cada función.
  - Determina si el punto pertenece a la primera función (\*), segunda (o), intersección (#), eje X (-), eje Y (|) o al origen (+).
- Construye un string multilinea con la representación ASCII.
- Agrega la leyenda al final.
- Inserta el resultado en el Text widget.

### 3.3 Función limpiar()

- Borra los valores de las entradas y del área de resultados.
- Mantiene el Text widget en estado deshabilitado.

### 3.4 Interfaz gráfica

- Título: “ Graficador ASCII de Funciones” con fuente Orbitron y color cian brillante.
- Marco de entradas:
  - Dos Entry para funciones.
  - Botón “Graficar”.
  - Botón “Limpiar”.
- Bind de teclado: tecla Enter ejecuta graficar.
- Marco de resultados:
  - LabelFrame con título “RESULTADOS”.
  - Caja de texto con scroll vertical que muestra la gráfica ASCII.

### 3.5 Estilos visuales

- Fondo oscuro espacial (#0B0C10).
- Botones futuristas: cian en reposo, verde azulado al presionar.
- Entrada con fondo oscuro, letras gris claro y cursor cian.
- Tipografía de estilo tecnológico para el título y consola de resultados.

### 4. Funcionamiento

1. El usuario escribe dos funciones matemáticas en los campos correspondientes.
2. Al presionar “Graficar”:
  - Cada función se prepara para evaluación correcta en Python.
  - Se genera la matriz de puntos y se dibuja en ASCII según las condiciones de intersección, ejes y origen.
  - Se muestra el resultado en el área de texto junto con la leyenda explicativa.
3. “Limpiar” reinicia la interfaz para nuevas funciones.

### 5. Ventajas

- Permite visualizar la interacción entre dos funciones en un plano cartesiano ASCII.
- Resalta intersecciones y ejes de forma clara.
- Interfaz atractiva y moderna.
- No requiere librerías externas de gráficos.

### 6. Limitaciones

- Representación gráfica limitada a caracteres ASCII; no permite zoom ni curvas suaves.
- No soporta funciones complejas como sin, cos o logaritmos.
- Rango de gráficos fijo (-20 a 20), puede no ajustarse a funciones muy grandes.
- Evaluación de expresiones depende de eval, lo que puede ser peligroso si se ingresa código malicioso.

### 7. Posibles mejoras

- Integrar soporte para funciones trigonométricas o exponenciales.

- Ajustar rango dinámicamente según la magnitud de las funciones.
- Permitir guardar la gráfica ASCII en un archivo .txt.
- Sustituir eval por sympy para mayor seguridad y soporte simbólico.
- Añadir colores distintos para cada función usando librerías como colorama.

## 8. Conclusión

El programa proporciona un graficador de funciones en ASCII educativo y visualmente atractivo, ideal para aprender relaciones entre funciones y sus intersecciones de manera sencilla. Su interfaz futurista mejora la experiencia de usuario y permite una interacción rápida con resultados inmediatos.