

Trabajo 2: Graficador de Funciones Lineales

Alumno: Ronald Wilder Incacutipa Muñuico

Docente: Ing. Torres Cruz Fred

Programación Numérica – FINESI
Universidad Nacional del Altiplano

Descripción del Problema

Se requiere un programa en Python que permita graficar **dos funciones lineales** ingresadas por el usuario. El sistema debe pedir además un valor de *paso*, con el cual se recorrerán los valores de x en el intervalo $[-1000, 1000]$.

El programa mostrará:

- La gráfica de las dos funciones en un plano cartesiano con ejes numerados.
- Los valores de las funciones evaluadas en cada x según el paso.

Especificaciones

- Entrada: dos expresiones lineales y el paso.
- Salida: gráfica de las funciones y listado de valores (x, y) .
- Restricciones: solo se aceptan funciones lineales en una variable x .

Ejemplo

$$f_1(x) = 2x, \quad f_2(x) = x$$

Con un paso de 10, el programa evaluará desde $x = -1000$ hasta $x = 1000$, mostrando y graficando los resultados.

Código en Python

```
1 import tkinter as tk
2
3 def evaluar(expr, x):
4     try:
5         return eval(expr, {"x": x})
6     except:
7         return None
8
9 def graficar():
10     f1, f2 = entrada1.get(), entrada2.get()
11     paso = int(entrada_paso.get())
12     canvas.delete("all")
13     salida.delete("1.0", tk.END)
14
15     w, h = 800, 600
16     cx, cy = w//2, h//2
17     esc = 30
18
19     canvas.create_line(0, cy, w, cy, fill="black")
20     canvas.create_line(cx, 0, cx, h, fill="black")
21
22     for i in range(-int(w/(2*esc)), int(w/(2*esc))+1):
23         x = cx + i*esc
24         canvas.create_line(x, cy-5, x, cy+5, fill="black")
25         canvas.create_text(x, cy+15, text=str(i), font=("Arial", 7))
26     for j in range(-int(h/(2*esc)), int(h/(2*esc))+1):
27         y = cy - j*esc
28         canvas.create_line(cx-5, y, cx+5, y, fill="black")
29         canvas.create_text(cx-15, y, text=str(j), font=("Arial", 7))
30
31     pts1 = []
32     salida.insert(tk.END, f"Valores de {f1}:\n")
33     for x in range(-1000, 1001, paso):
34         y = evaluar(f1, x)
35         if y is not None:
36             px, py = cx + x*esc/10, cy - y*esc/10
37             pts1.append((px, py))
38             salida.insert(tk.END, f"x={x}, y={y}\n")
39     for i in range(len(pts1)-1):
40         canvas.create_line(pts1[i], pts1[i+1], fill="red")
41
42
43     pts2 = []
44     salida.insert(tk.END, f"\nValores de {f2}:\n")
45     for x in range(-1000, 1001, paso):
46         y = evaluar(f2, x)
47         if y is not None:
48             px, py = cx + x*esc/10, cy - y*esc/10
49             pts2.append((px, py))
```

```

50         salida.insert(tk.END, f"x={x}, y={y}\n")
51     for i in range(len(pts2)-1):
52         canvas.create_line(pts2[i], pts2[i+1], fill="blue")
53
54
55 root = tk.Tk()
56 root.title("Graficador Lineal")
57
58 tk.Label(root, text="Función 1 (ej: 2*x+3):").pack()
59 entrada1 = tk.Entry(root); entrada1.pack()
60 tk.Label(root, text="Función 2 (ej: -3*x/4+2):").pack()
61 entrada2 = tk.Entry(root); entrada2.pack()
62 tk.Label(root, text="Paso (ej: 10):").pack()
63 entrada_paso = tk.Entry(root); entrada_paso.insert(0,"10"); entrada_paso.pack()
64
65 tk.Button(root, text="Graficar", command=graficar).pack()
66
67 canvas = tk.Canvas(root, width=800, height=600, bg="white"); canvas.pack(side="left")
68 salida = tk.Text(root, width=40, height=35); salida.pack(side="right")
69
70 root.mainloop()

```

Resultados

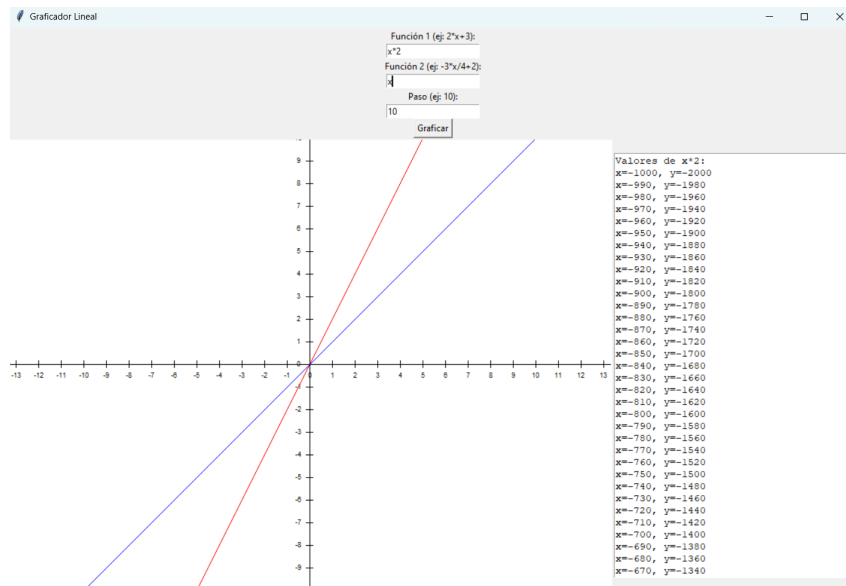


Figure 1: Captura de pantalla del graficador lineal mostrando dos funciones.

Conclusión

El programa permite graficar funciones lineales de manera sencilla en una ventana gráfica con Tkinter. Se ingresan dos funciones y un paso, y el sistema devuelve tanto la representación visual como los valores numéricos evaluados en el intervalo $[-1000, 1000]$. Esto facilita la comprensión del comportamiento de funciones lineales y su comparación.