

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERIA ESTADISTICA E INFORMATICA



RECONOCIMIENTO DE FUNCIONES

Autores:

Aquino Sandoval Jean Carlos (214689)

Semestre: IV grupo: A

Docente: Ing. Torres Cruz Fred

Area: Programacion Numerica

Puno – Perú

2025

Reconocimiento de Funciones

1. Objetivo

Desarrollar una herramienta gráfica interactiva que permita analizar expresiones matemáticas ingresadas por el usuario, identificando las variables involucradas, contando el número de operaciones y mostrando resultados de manera clara y visual. Se busca combinar funcionalidad matemática con una interfaz de estilo futurista, adecuada para entornos educativos o presentaciones interactivas.

2. Descripción general del código

El programa está desarrollado en Python usando la biblioteca Tkinter y ttk para los elementos de la interfaz gráfica. Permite:

1. Ingresar una función matemática en una caja de texto.
2. Analizar la expresión para identificar:
 - Variables presentes.
 - Número de variables.
 - Número de operaciones matemáticas (+, -, *, /, ^), incluyendo multiplicación implícita como 2x o xy.
3. Mostrar los resultados en un área de texto con scroll.
4. Limpiar los campos de entrada y resultados mediante un botón dedicado.

3. Estructura y secciones del código

3.1 Función analizar(event=None)


- Obtiene la expresión ingresada por el usuario desde la caja de entrada (entrada.get()).
- Verifica si la expresión no está vacía; de lo contrario muestra una advertencia.
- Recorre cada carácter de la expresión:
 - Si es letra, se considera una variable y se agrega a un conjunto (set).
 - Si es operador (+, -, *, /, ^), se incrementa el contador de operaciones.
 - Detecta multiplicación implícita:

- Entre un número y una variable (2x).
- Entre dos variables consecutivas (xy).
- Genera un texto de resultados con los valores analizados.
- Muestra el resultado en un Text widget, configurado como solo lectura.

3.2 Función limpiar()

- Borra la entrada de la función.
- Limpia el área de resultados.
- Mantiene la caja de resultados deshabilitada para solo lectura.

3.3 Configuración de la ventana principal

- Título: “ Analizador Matemático Avanzado”.
- Tamaño fijo de 600x400 píxeles.
- Fondo oscuro (#0B0C10) con estilo futurista.
- No permite redimensionar la ventana.

3.4 Estilos personalizados

- Se utiliza ttk.Style() para definir:
 - Botones con fuente “Segoe UI Semibold”, color de fondo oscuro y efecto al presionar (#45A29E).
 - Caja de entrada con fondo oscuro, color de letra gris claro y cursor color cian (#66FCF1).
- Tema clam para compatibilidad de estilos.

3.5 Interfaz gráfica

- **Título principal:** en fuente “Orbitron”, color cian brillante.
- **Marco de entrada:** contiene:
 - Caja de texto para ingresar la función.
 - Botón “Analizar”.
 - Botón “Limpiar”.
- **Bind de teclado:** presionar Enter ejecuta automáticamente la función analizar.
- **Marco de resultados:**
 - LabelFrame con título “RESULTADOS”.

- Caja de texto con scroll vertical.
- Presenta los resultados de forma legible y organizada.

4. Funcionamiento

1. El usuario escribe una expresión matemática en la caja de entrada (por ejemplo: $2x + y - z^2$).
2. Al presionar “Analizar”:
 - Se detectan las variables: x, y, z.
 - Se cuentan las operaciones: +, -, *, ^, y multiplicaciones implícitas.
 - Se muestra un resumen en la sección de resultados.
3. El botón “Limpiar” permite reiniciar la entrada y el área de resultados.

5. Ventajas

- Interfaz intuitiva y visualmente atractiva.
- Manejo de multiplicación implícita.
- Resultado organizado y fácil de interpretar.
- Compatible con expresiones matemáticas básicas.

6. Limitaciones

- No evalúa funciones matemáticas avanzadas (sin, cos, log) de forma automática.
- No valida completamente la sintaxis de la expresión; errores complejos pueden causar resultados inesperados.
- Solo analiza la estructura de la expresión, no su valor numérico.

7. Posibles mejoras

- Agregar compatibilidad con funciones matemáticas estándar (math.sin, math.exp, etc.).
- Mostrar resultados gráficos de la función ingresada en un Canvas.
- Añadir etiquetas dinámicas para cada variable detectada.
- Mejorar la detección de errores en expresiones complejas.

8. Conclusión

Este código constituye una herramienta educativa efectiva para el **análisis sintáctico de expresiones matemáticas**, combinando funcionalidad y estética

visual. Su diseño en Tkinter permite futuras ampliaciones, como integración con gráficos y cálculo de valores numéricos, manteniendo un estilo futurista atractivo para estudiantes y docentes.