

Cálculo y representación gráfica de una ecuación cuadrática en Python

[crayon-6a9d6063e1b5d800571267/]

Explicación del código

1. Función calcular_vértice:
2. Definición de Coeficientes:
3. Generación de Datos:
4. Creación del Gráfico:
5. Impresión en Consola:

```
import numpy as np
import matplotlib.pyplot as plt

def calcular_vértice(a, b, c):
    """Calcula el vértice de la parábola para la ecuación  $ax^2 + bx + c$ ."""
    x_vértice = -b / (2 * a)
    y_vértice = a * x_vértice**2 + b * x_vértice + c
    return x_vértice, y_vértice

# Coeficientes de la ecuación cuadrática
a = 1
b = -4
c = 3

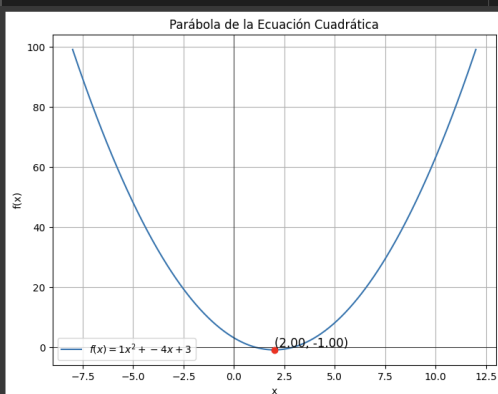
# Calcular el vértice
x_vértice, y_vértice = calcular_vértice(a, b, c)

# Definir el rango de x
x = np.linspace(x_vértice - 10, x_vértice + 10, 400)
y = a * x**2 + b * x + c

# Crear la figura y el eje
plt.figure(figsize=(8, 6))
plt.plot(x, y, label=f'sf(x) = {a}x^2 + {b}x + {c}')
plt.scatter([x_vértice], [y_vértice], color='red', zorder=5)
plt.text(x_vértice, y_vértice, f'({x_vértice:.2f}, {y_vértice:.2f})', fontsize=12, verticalalignment='bottom')

# Configurar el gráfico
plt.title('Parábola de la Ecuación Cuadrática')
plt.xlabel('x')
plt.ylabel('f(x)')
plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)
plt.grid(True, which='both')
plt.legend()
plt.show()

# Imprimir el vértice en la consola
print(f"El vértice de la parábola es ({x_vértice:.2f}, {y_vértice:.2f})")
```



El vértice de la parábola es (2.00, -1.00)

Parábola de la Ecuación Cuadrática

