Instituto Tecnológico Superior Progreso

Tarea: 1.1
Tema: Introducción a los métodos numéricos

Asignatura:	____________________	**Calificación:**	________
Profesor:	Dr. Iván de Jesús May-Cen	**Fecha:**	________
ESTUDIANTE:	____________________	**Carrera:**	________

I. OBJETIVO:

El estudiante implementará programas en Python para calcular la precisión de representaciones numéricas, así como el error absoluto, el error relativo y error cuadrático en diferentes contextos.

II. INSTRUCCIONES:

Resuelve los siguientes ejercicios programando en **Python**. Entrega el código junto con un informe breve que explique los resultados obtenidos.

III. ENTREGA:

- Código fuente de los programas en Python.
- Informe breve con:
  - Explicación de los resultados.
  - Tablas y gráficas obtenidas.
  - Análisis de los errores encontrados.

IV. EJERCICIOS:

Ejercicio 1: Precisión de la representación numérica

Escribe un programa que determine la **precisión de máquina** (ε) en punto flotante. Puedes hacerlo usando el siguiente método:

1. Iniciar con un valor de $\varepsilon = 1.0$.
2. Dividirlo entre 2 en cada iteración.
3. Comprobar en qué punto $1.0 + \varepsilon = 1.0$ en la representación de la computadora.

Salida esperada: El valor de la precisión de máquina en el sistema utilizado.

Código: `https://github.com/imaycen/ejercicio1-Precision`

Ejercicio 2: Cálculo del error absoluto, relativo y cuadrático en una aproximación

Se desea calcular el valor de π utilizando la aproximación de Leibniz:

$$\pi \approx 4 \sum_{n=0}^{N} \frac{(-1)^n}{2n+1} \tag{1}$$

1. Implementa este método en un programa y aproxima π usando diferentes valores de N (por ejemplo, $N = 10, 100, 1000, 10000$).

2. Calcula el **error absoluto**, el **error relativo** y el **error cuadrático** en cada caso usando el valor real de $\pi \approx 3.141592653589793$.

3. Representa gráficamente el error absoluto y el error relativo en función del número de términos N.

Salida esperada: Una tabla y una gráfica donde se observe cómo disminuyen los errores al incrementar N.

Código: `https://github.com/inaycen/ejercicio2-Leibniz`

Ejercicio 3: Errores en operaciones numéricas

1. Escribe un programa que realice la resta de dos números cercanos entre sí:

   $$x = 1.0000001, \quad y = 1.0000000 \tag{2}$$

   Calcula la diferencia y determina el **error absoluto** y **error relativo** si el valor exacto esperado es 0.0000001.

2. Repite el ejercicio usando números mucho más pequeños:

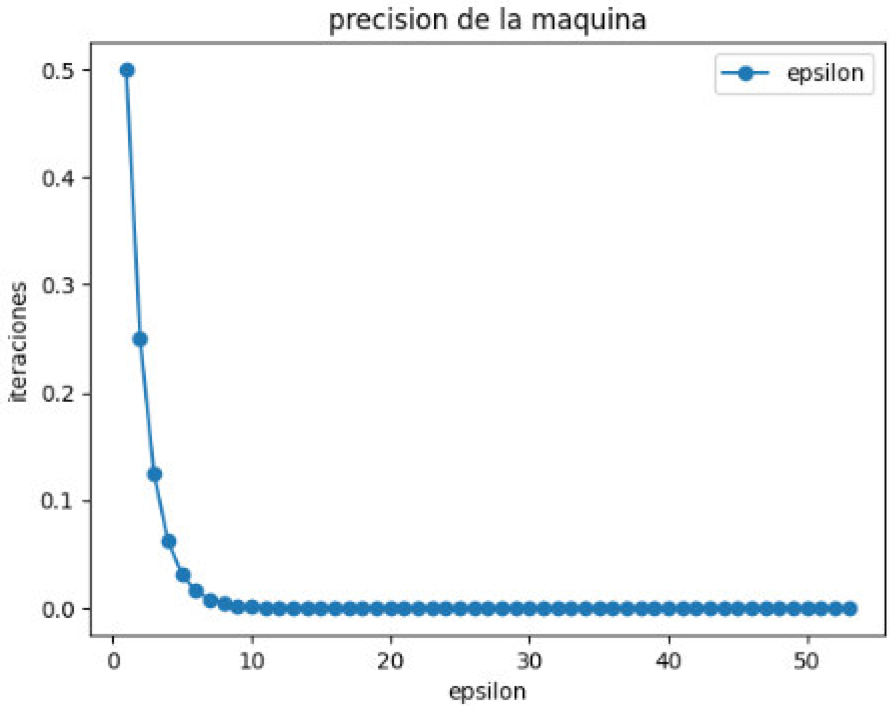
   $$x = 1.000000000000001, \quad y = 1.000000000000000 \tag{3}$$

   ¿Cómo afecta la precisión numérica en cada caso?

Salida esperada: Un análisis del impacto de la precisión numérica en la resta de números cercanos.

Código: `https://github.com/inaycen/ejercicio3-Operaciones`

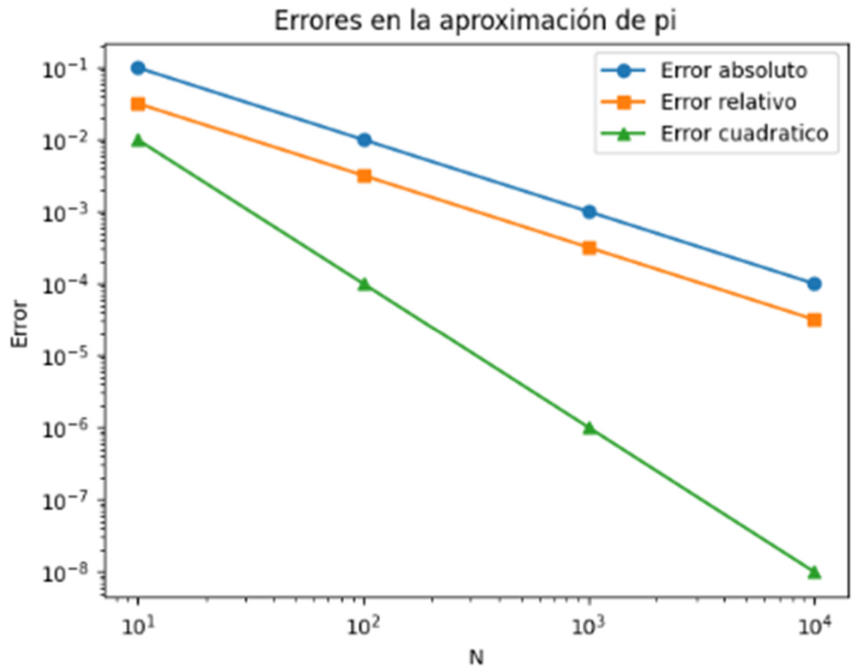
REPORTE DEL CODIGO #1



La gráfica muestra cómo la precisión numérica de las computadoras disminuye al dividir épsilon entre 2 en cada iteración, reflejando el límite de precisión que tienen las máquinas.

NUMERO DE ITERACION	PRECISION DE LA MAQUINA
1	0.5
10	0.0009765625
20	9.5367431640625e-07
30	9.313225746154785e-10
40	9.094947017729282e-13
50	8.881784197001252e-16
51	4.440892098500626e-16
52	2.220446049250313e-16
53	1.1102230246251565e-16

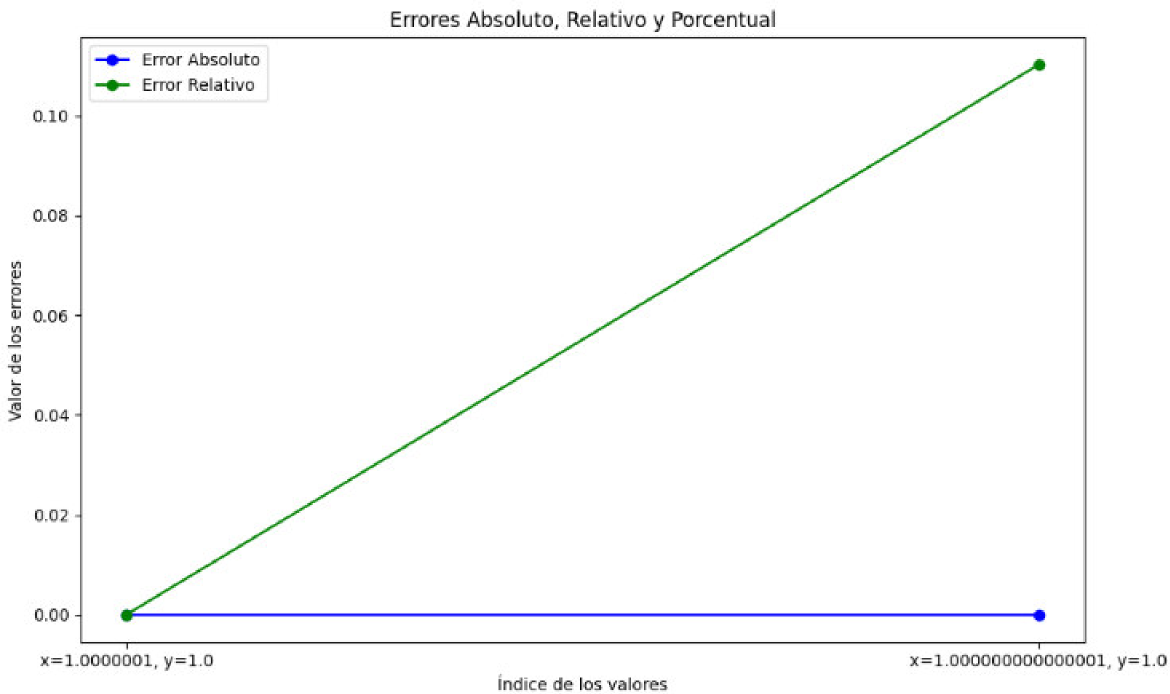
REPORTE DEL CODIGO 2



La gráfica muestra cómo disminuyen los errores (absoluto, relativo y cuadrático) al aproximar π usando la serie de Leibniz con diferentes valores de N. A medida que NNN aumenta, los errores se reducen, pero la serie converge lentamente. Los errores se muestran en una escala logarítmica para resaltar la disminución con el aumento de términos. Se observa que el **error absoluto** y **relativo** disminuyen conforme N crece. En general, la gráfica ilustra la mejora en la aproximación de π a medida que se agregan más términos.

N_VALUES	ERROR ABSOLUTO	ERROR RELATIVO	ERROR CUADRATICO
N=10	0.09975303466038987	0.03175237710923643	0.009950667923956944
N=100	0.00999975003123943	0.0031830192943 1018	9.999500068727297e-05
N=1000	0.000999999749998981	0.0003183098066059948	9.999994999980246e-07
N=10000	9.99999997586265e-05	3.18309885415475e-05	9.9999999517253e-09

REPORTE DEL CODIGO 3



El código calcula los errores absoluto, relativo y porcentual entre dos valores aproximados x y y con respecto a un valor real. La función calcular_errores realiza los cálculos y luego los imprime. Se usan dos ejemplos de valores para ilustrar cómo varían estos errores. Los errores se almacenan en listas y se grafican utilizando matplotlib en un gráfico de líneas. En el gráfico, el eje X representa los pares de valores x y y, mientras que el eje Y muestra los errores correspondientes. Esto permite comparar visualmente los tres tipos de errores para los diferentes pares de valores.

Para x=1.0000001, y=1.0:	
DIFERENCIA	1.0000000005838672e-07
ERROR ABSOLUTO	5.838672220806777e-17
ERROR RELATIVO	5.838672220806777e-10
ERROR PORCENTUAL	5.838672220806777e-08%
Para x=1.000000000000001, y=1.0:	
DIFERENCIA	1.1102230246251565e-15
ERROR ABSOLUTO	1.1022302462515646e-16
ERROR RELATIVO	0.11022302462515646
ERROR PORCENTUAL	11.022302462515645%