

Investigación sobre Logaritmos y Antilogaritmos

Para comprender cómo funcionan los logaritmos y antilogaritmos, comenzamos con un ejemplo básico:

Ejemplo Matemático:

Para calcular el logaritmo base 10 de 100, utilizamos la fórmula:

$$\log_{10}(100) = y$$

Sabemos que ($10^2 = 100$), por lo tanto, ($y = 2$).

Ahora, para encontrar el antilogaritmo de (y) en base 10, utilizamos:

$$10^y = x$$

Sustituyendo (y) por 2, obtenemos:

$$10^2 = 100$$

Por lo tanto, ($x = 100$), lo que demuestra que el antilogaritmo de 2 en base 10 es 100.

Cálculo de Logaritmos

Forma Manual (Ruda)

Para calcular logaritmos de manera manual sin utilizar librerías especializadas, podemos implementar una función básica en Python:

```
def logaritmo_basico(x):  
    if x <= 0:  
        return "Error: x debe ser mayor que 0"  
    n = 1000000 # Aumentar para mayor precisión  
    paso = (x - 1) / n  
    log_approx = 0  
    for i in range(1, n + 1):  
        log_approx += paso / (1 + paso * (i - 1))  
    return log_approx
```

```
# Prueba con un número
```

```
x = 2.71828 # Aproximadamente e
```

```
log_resultado = logaritmo_basico(x)
print(log_resultado)
```

Uso de la Librería `math`

Para una solución más precisa y sencilla, podemos utilizar la librería `math` de Python:

```
import math

# Logaritmo base 10 de 100
resultado = math.log(100, 10)
print(resultado) # Esto debería imprimir 2.0
```

Cálculo de Antilogaritmos

Forma Manual (Ruda)

De manera similar, para calcular antilogaritmos de forma manual, podemos definir una función que eleve una base a un exponente dado

```
def antilogaritmo(base, exponente):
    return base ** exponente

# Ejemplo de uso
base = 10
exponente = 3
resultado = antilogaritmo(base, exponente)
print(f"El antilogaritmo en base {base} de {exponente} es {resultado}")
```

Uso de la Librería `math`

Para calcular antilogaritmos utilizando `math`, hacemos uso de la función `pow` para elevar una base a un exponente:

```
import math

# Antilogaritmo de 2 en base 10
resultado_antilog = math.pow(10, 2)
```

```
print(resultado_antilog) # Esto debería imprimir 100.0
```

Potencia y raíz

Teniendo en cuenta que la raíz cuadrada es igual a elevar a la fracción de esta misma:

$$\sqrt{3} = 3^{1/2}$$

Tenemos:

```
# Potencias y raíces

y=9 # Valor
pow=1/2 # Potencia

x = y**(pow) # Potenciar un número

print(x) #Imprimimos el resultado
```

REFERENCIAS

[LINK DE REREFENCIA](#)