

Actividad extracurricular 09 - Función atan2

Nombre: Alexis Bautista

Fecha de entrega: 17 de diciembre del 2024

Paralelo: GR1CC

Enlace de GitHub: <https://github.com/alexis-bautista/Actividad09-MN.git>

Función atan2

La función matemática `atan2`, permite calcular el ángulo cuyo valor tangente es igual a un cociente dado. Es fundamental en trigonometría para resolver problemas de triángulos rectángulos y convertir entre coordenadas cartesianas y polares. Su fórmula general es $\arctan(y/x)$, pero la función `atan2` extiende esta funcionalidad considerando los signos de las coordenadas x e y lo que asegura la correcta determinación del cuadrante del ángulo.

En programación, `atan2` es ampliamente utilizada en áreas como la robótica, gráficos por computadora y análisis vectorial. Los resultados de esta función se expresan en radianes, y su implementación en bibliotecas especializadas garantiza precisión y eficiencia.

Ejemplos de representaciones comunes incluyen:

$$\theta = \text{atan2}(y, x)$$

Donde:

- y y x representan las coordenadas cartesianas del punto.
- θ es el ángulo resultante, medido en radianes.

¿Por qué se recomienda usar la función atan2?

La función `atan2` se recomienda debido a su capacidad para manejar casos complejos de cálculo angular, incluyendo:

- **Determinación precisa del cuadrante angular**, lo que es crítico en aplicaciones tridimensionales y en la rotación de vectores.
- **Conversión de coordenadas cartesianas a polares**, lo cual facilita el análisis en sistemas de referencia no cartesianas.
- **Estabilidad en aplicaciones prácticas**, como cálculos relacionados con trayectorias, navegación y matrices de rotación en sistemas embebidos.

Además, su implementación en software es estándar en lenguajes como Python, C++ y Java, ofreciendo resultados precisos para tareas técnicas y científicas.

Diferencias entre función atan y atan2

La función `atan` calcula el arco tangente de un número y devuelve un ángulo en radianes dentro del intervalo $-\pi/2$ a $\pi/2$. Sin embargo, esta función asume un único cuadrante y no considera los signos de x e y , lo que limita su precisión en aplicaciones complejas.

Por otro lado, la función `atan2` supera esta limitación al evaluar directamente las coordenadas (x, y) , determinando el cuadrante correcto del ángulo resultante. Esto la hace ideal para trabajar con sistemas vectoriales, gráficos 2D/3D y modelos matemáticos donde los signos de las coordenadas son esenciales.

Ejemplos en Python

La función `atan2` en Python, disponible a través del módulo `math` como `math.atan2(y, x)`, se utiliza para calcular el ángulo (en radianes) formado por un vector respecto al eje X positivo en un sistema de coordenadas cartesianas. Se usa en aplicaciones que involucran trigonometría, vectores y transformaciones de coordenadas

1. Conversión de coordenadas cartesianas a polares

In [1]: `import math`

```
x, y = 5, 5 # Coordenadas cartesianas
r = math.sqrt(x**2 + y**2) # Magnitud del vector
theta = math.atan2(y, x) # Ángulo en radianes
print(f"Coordenadas polares: r = {r}, θ = {theta} radianes.")
```

Coordenadas polares: r = 7.0710678118654755, θ = 0.7853981633974483 radianes.

2. Determinación de la dirección de un punto relativo a otro

In [2]: `x1, y1 = 2, 3 # Punto inicial`
`x2, y2 = 7, 8 # Punto final`
`angulo = math.atan2(y2 - y1, x2 - x1)`
`print(f"La dirección del punto ({x2}, {y2}) respecto a ({x1}, {y1}) es {angulo}")`

La dirección del punto (7, 8) respecto a (2, 3) es 0.7853981633974483 radianes.

3. Detección del cuadrante de un punto

In [4]: `x, y = -4, 7 # Coordenadas de un punto`
`angulo = math.atan2(y, x)`

`if 0 <= angulo < math.pi/2:`
 `cuadrante = "primer cuadrante"`
`elif math.pi/2 <= angulo <= math.pi:`
 `cuadrante = "segundo cuadrante"`
`elif -math.pi <= angulo < -math.pi/2:`
 `cuadrante = "tercer cuadrante"`
`else:`
 `cuadrante = "cuarto cuadrante"`

```
print(f"El punto ({x}, {y}) está en el {cuadrante}, con un ángulo de {angulo} ra
```

El punto (-4, 7) está en el segundo cuadrante, con un ángulo de 2.0899424410414196 radianes.

4. Simulación de un objeto que apunta hacia un objetivo

```
In [7]: # Coordenadas del objeto y del objetivo
objeto_x, objeto_y = 2, 2
objetivo_x, objetivo_y = 8, 6

# Calcular el ángulo de orientación
angulo_hacia_objetivo = math.atan2(objetivo_y - objeto_y, objetivo_x - objeto_x)
angulo_grados = math.degrees(angulo_hacia_objetivo) # Convertir a grados para m

print(f"El ángulo para apuntar al objetivo es {angulo_hacia_objetivo:.2f} radian
```

El ángulo para apuntar al objetivo es 0.59 radianes (33.69 grados).

5. Cálculo del ángulo entre dos vectores en 2D

```
In [10]: # Coordenadas de Los vectores
vector_a = (3, 4) # (x1, y1)
vector_b = (6, 2) # (x2, y2)

# Componentes de Los vectores
x1, y1 = vector_a
x2, y2 = vector_b

# Calcular el ángulo entre Los vectores
angulo_a = math.atan2(y1, x1)
angulo_b = math.atan2(y2, x2)
angulo_entre = angulo_b - angulo_a

# Asegurar que el ángulo esté en el rango [-pi, pi]
angulo_entre = (angulo_entre + math.pi) % (2 * math.pi) - math.pi

print(f"El ángulo entre los vectores {vector_a} y {vector_b} es {angulo_entre:.2
```

El ángulo entre los vectores (3, 4) y (6, 2) es -0.61 radianes.

Bibliografía

- Sandoval-Gutiérrez, J., Herrera-Lozada, J. C., Laguna-Sánchez, G. A., & Álvarez Cedillo, J. A. (2023). Análisis numérico de atan2 () para sistemas embebidos. Computación y Sistemas, 27(1), 79-87. recuperado de: <https://www.cys.cic.ipn.mx/ojs/index.php/CyS/article/view/4141/3498>
- Ortiz, A. (2024, abril 27). Mastering the arctan formula: A comprehensive guide for mathematical enthusiasts. Apolonio.es. <https://apolonio.es/arctan-formula/>
- IBM Integration Bus 10.0.0. (2023, junio 21). Ibm.com. <https://www.ibm.com/docs/es/integration-bus/10.0?topic=functions-atan2-function>
- Función ATAN. (s/f). Microsoft.com. Recuperado el 15 de diciembre de 2024, de <https://support.microsoft.com/es-es/office/funci%C3%B3n-atan-50746fa8-630a-406b-81d0-4a2aed395543#:~:text=Descripci%C3%B3n,/2%20a%20pi/2>.

