gerar a sequência de Fibonacci usando recursão

Para gerar a sequência de Fibonacci usando recursão em Python, você pode definir uma função recursiva que calcula o valor de Fibonacci para um dado índice. A sequência de Fibonacci é definida como:

* F(0)=0F(0) = 0F(0)=0
* F(1)=1F(1) = 1F(1)=1
* F(n)=F(n−1)+F(n−2)F(n) = F(n-1) + F(n-2)F(n)=F(n−1)+F(n−2) para n>1n > 1n>1

Aqui está um exemplo de como implementar isso em Python:

python

Copiar código

def fibonacci(n):

if n == 0:

return 0

elif n == 1:

return 1

else:

return fibonacci(n - 1) + fibonacci(n - 2)

# Exemplo de uso: Gerando os primeiros 10 números da sequência de Fibonacci

for i in range(10):

print(fibonacci(i))

**Explicação:**

1. **Função fibonacci(n)**: Esta função calcula o número de Fibonacci para o índice n.
   * Se n é 0, retorna 0 (caso base).
   * Se n é 1, retorna 1 (caso base).
   * Para n maior que 1, a função chama a si mesma recursivamente para calcular os dois números anteriores na sequência, somando-os para obter o valor atual.
2. **Exemplo de uso**: Um loop for é usado para gerar e imprimir os primeiros 10 números da sequência de Fibonacci.

**Considerações de Desempenho**

A abordagem recursiva é simples e intuitiva, mas pode ser ineficiente para números grandes, pois recalcula os mesmos valores múltiplas vezes. Para melhorar a eficiência, pode-se usar memoização ou a abordagem iterativa.