calcular o fatorial de um número usando recursão

Para calcular o fatorial de um número usando recursão em Python, você pode definir uma função recursiva que multiplica o número pelo fatorial do número anterior até chegar a 1. O fatorial de um número nnn é definido como:

* n!=n×(n−1)×(n−2)×…×1n! = n \times (n-1) \times (n-2) \times \ldots \times 1n!=n×(n−1)×(n−2)×…×1
* Com o caso base sendo 0!=10! = 10!=1 e 1!=11! = 11!=1

Aqui está uma implementação em Python:

python

Copiar código

def fatorial(n):

# Caso base: 0! e 1! são iguais a 1

if n == 0 or n == 1:

return 1

else:

# Chamada recursiva: n \* (n-1)!

return n \* fatorial(n - 1)

# Exemplo de uso: Calculando o fatorial de 5

resultado = fatorial(5)

print(f"O fatorial de 5 é: {resultado}")

**Explicação:**

1. **Função fatorial(n)**: Esta função calcula o fatorial de n.
   * **Caso base**: Se n é 0 ou 1, retorna 1, pois 0!=10! = 10!=1 e 1!=11! = 11!=1.
   * **Chamada recursiva**: Se n é maior que 1, a função retorna n multiplicado pelo fatorial de n-1 (ou seja, n \* fatorial(n - 1)), até que atinja o caso base.
2. **Exemplo de uso**: A função é chamada com o valor 5, e o resultado é o fatorial de 5, que é 120.

A recursão simplifica a implementação do cálculo do fatorial, mas para números muito grandes, pode ser mais eficiente usar uma abordagem iterativa ou bibliotecas especializadas devido às limitações de profundidade da pilha de chamadas em muitas linguagens de programação, incluindo Python.