

Fundamentos de Algoritmos

Recursividade

Marco A L Barbosa

malbarbo.pro.br

Lembre-se de seguir o processo de projeto de funções e de usar o `mypy` e o `doctest` na etapa de verificação.

Começando

- 1) O que é uma definição recursiva? O que é preciso para que uma definição recursiva seja bem formada?
- 2) O que é uma função recursiva? O que é preciso para que uma função recursiva seja bem formada?
- 3) Quais os dois aspectos que precisamos considerar no projeto de funções recursivas?
- 4) Qual é a ideia da estratégia de diminuição e conquista?
- 5) Quando não podemos utilizar a estratégia de diminuição e conquista?

Praticando

- 6) Projete uma função recursiva que receba como parâmetro um número natural n e um valor v e crie uma nova lista com n repetições do valor v .
- 7) Projete uma função recursiva que receba como parâmetro um número natural n e calcule o produto dos números $1, 2, \dots, n$.
- 8) Projete uma função recursiva que receba como parâmetro um número natural n e gere uma string com os números de 1 até n separados por vírgula. Por exemplo, para $n = 3$, a função deve gerar "1,2,3".
- 9) Recursão indireta é quando duas ou mais funções chamam uma a outra. Defina duas funções `impar` e `par`, uma em termos da outra, isto é, a função `impar` deve chamar a função `par` e a função `par` deve chamar a função `impar` (a recursão para no caso base).

Para os exercícios a seguir, use diminuição física do arranjo.

- 10) Projete uma função recursiva que concatene todas as strings de uma lista.
- 11) Projete uma função recursiva que determine se um número está em uma lista de números.
- 12) Uma lista de números é chamada de lista binária se todos os seus elementos são 0 ou 1. Projete uma função recursiva que verifique se uma lista é binária.
- 13) Projete uma função recursiva que encontre o valor máximo de uma lista não vazia. Dica: mude o caso base para uma lista com um elemento.

Para os exercícios a seguir, use diminuição lógica do arranjo decrementando o tamanho a partir do final.

- 14) Projete uma função recursiva que encontre o tamanho máximo entre todas as strings de uma lista de strings.
- 15) Projete uma função recursiva que crie uma nova lista com todos os valores positivos de uma lista de números.
- 16) Projete uma função recursiva que altere uma lista somando 1 em cada um dos seus elementos.

Para os exercícios a seguir, use a estratégia de diminuição que for mais adequada.

- 17) (Desafio) Projete uma função recursiva que verifique se uma lista é palíndromo. Dica: diminua do início e do final.
- 18) (Desafio) Projete uma função recursiva que altere uma lista invertendo a ordem dos seus elementos
- 19) (Desafio) Projete uma função recursiva que encontre o tamanho da maior sublista de zeros consecutivos de uma dada lista de inteiros. Por exemplo, para a lista [0, 0, 2, 1, 0, 0, 0, 7], a resposta é 3, e para a lista [0, 0, 1, 0, 4] a resposta é 2.