

Recursão

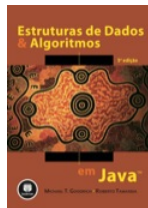
Roland Teodorowitsch

Algoritmos e Estruturas de Dados I - Escola Politécnica - PUCRS

10 de agosto de 2023

Recursão

Leitura(s) Recomendada(s)



Seção 3.5

GOODRICH, Michael T.; TAMASSIA, Roberto. **Estruturas de dados e algoritmos em Java**. Tradução: Bernardo Copstein. 5. ed. Porto Alegre: Bookman, 2013. xxii, 713 p. E-book. ISBN 9788582600191. Tradução de: Data Structures and Algorithms in Java, 5th Edition. Disponível em: <<https://integrada.minhabiblioteca.com.br/#/books/9788582600191/>>. Acesso em: 01 ago. 2023.

Algoritmo recursivo

- É um algoritmo que faz chamada a si próprio dentro de sua implementação
- **Importante:** o algoritmo deve garantir que a recursão termine
 - Do contrário se entrará em um laço infinito
 - Importante considerar a condição de termino logo no início da implementação do algoritmo

Exemplo clássico: fatorial

- O fatorial de n é normalmente definido como:

$$n! = \prod_{k=1}^n k = n \times (n-1) \times (n-2) \times \dots \times 3 \times 2 \times 1, \quad \forall n \in \mathbb{N}$$

- Por exemplo:

$$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$$

- Mas também pode-se usar a sua definição recursiva:

$$n! = \begin{cases} 1, & \text{se } n = 0 \text{ ou } n = 1 \\ n \times (n-1)!, & \text{se } n > 1 \end{cases}$$

Implementação do fatorial

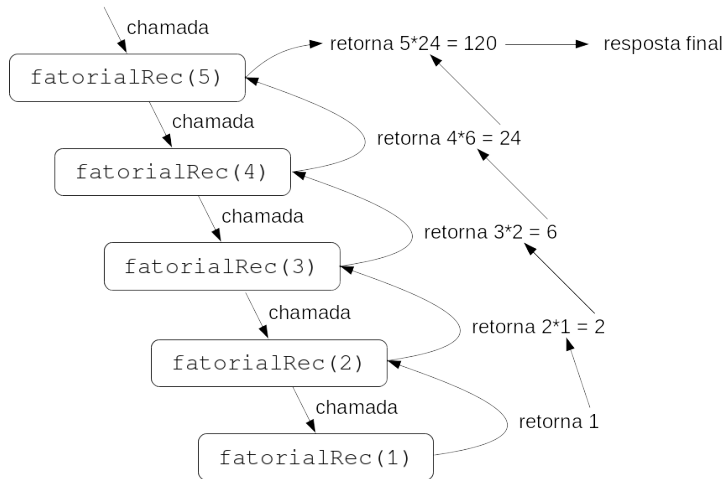
- Iterativo

```
unsigned int fatorial(unsigned int n) {  
    unsigned int res = 1;  
    for (unsigned int i=2; i<=n; ++i)  
        res *= i;  
    return res;  
}
```

- Recursivo

```
unsigned int fatorialRec(unsigned int n) {  
    if (n <= 1) return 1;  
    return n * fatorialRec(n-1);  
}
```

Rastreamento Recursivo



Recursividade

- Poderosa ferramenta de programação
- Apesar de bastante empregada, nem sempre ela deve ser aplicada
 - É preciso analisar o problema e ver se necessita de uma solução recursiva
- Quando bem empregada pode tornar a solução de um problema clara, simples e consisa

...

● ...

Vantagens

- Rotinas mais concisas
- Relação direta com uma prova por indução matemática
 - Indução matemática: metodo de prova matemática usado para demonstrar a verdade de um número infinito de proposições
 - Válida se funciona para n igual a 0 ou 1
 - Válida se vale para n igual a k e $k + 1$
 - Facilita verificar a correção

Desvantagens

- Cada chamada recursiva implica em um custo (tempo e espaço)
 - Informações são armazenadas na pilha
 - Para cada chamada realizada, um conjunto de variáveis locais é alocado (criado)
 - Cada chamada requer
 - O empilhamento de parâmetros e endereços de retorno da função que chama
 - O desempilhamento de parâmetros pela função que executa
 - Cada retorno requer
 - O desempilhamento do endereço de retorno
 - O empilhamento do resultado (ou passagem por registrador)
 - O desempilhamento do retorno pela função que chamou

Exercícios

- ❶ Faça o algoritmo da potência de forma não recursiva e de forma recursiva. Considere que o valor da base e do expoente são recebidos por parâmetro, inteiros e positivos.
- ❷ Faça um algoritmo recursivo que inverta a ordem dos elementos de um vetor.
- ❸ Faça um algoritmo recursivo para somar os elementos de um vetor.
- ❹ **Desafio:**
Faça um algoritmo recursivo para verificar se uma cadeia de caracteres é um palíndromo. Teste: “socorrammesubinoonibusemmarrocoss”.

Créditos

Créditos

- Estas lâminas contêm trechos de materiais criados e disponibilizados pelo professor Iłaniski Weber.