Algoritmos recursivos

Técnicas de Programação - 2025/02

Última atualização: 14/09/2025 19:05

Agradecimentos

Essa aula foi bastante inspirada nas do prof. Marcelo Hashimoto, que dá aulas de algoritmos na EngComp. Se tem alguma coisa ruim, deve ter sido alguma adição que fiz.

Slides

Use esta página para anotar os exemplos 1 e 2 dos slides.

Simulação

Agora vamos simular nós mesmos alguns exemplos. Queremos fazer isso para entender o **mecanismo da recursão**.

> Como assim?
>
> Simular não é necessariamente útil para encontrar problemas em algoritmos recursivos. Porém, neste momento queremos entender **como** a recursão acontece. Para facilitar um pouco, vamos usar código *C* nesta aula.

```c
int f(int n) {
    if (n == 1) {
        return 1;
    }
    int r = f(n - 1) + 1;
    return r;
}
```

```c
int bin(int n) {
    if (n == 0) {
        return 1;
    }
    int r = 2 * bin(n - 1);
    return r;
}
```

Uma ideia melhor

Como vimos acima, simular algoritmos recursivos é um processo lento, confuso e trabalhoso. Seria melhor se tivéssemos uma maneira de entender algoritmos recursivos sem precisar de tudo isso.

1. Para quais entradas não precisamos fazer chamada recursiva?
2. O programa termina?
3. O programa funciona, supondo que as chamadas recursivas retornam o valor correto?

Vamos praticar fazer essas duas perguntas para alguns algoritmos e com isso encontrar problemas e suas soluções.

```
void acumula(int v[], int n) {
    if (n == 1) {
        return;
    }
    acumula(v, n - 1);
    v[n - 1] += v[n - 2];
}
```

Exercício: Para quais entradas não precisamos fazer chamada recursiva?

Exercício: O programa termina?

Exercício: O programa funciona, supondo que as chamadas recursivas retornam o valor correto?

O programa abaixo recebe um array `v` e dois números `l` e `r`. Ele checa se o array é simétrico no intervalo de elementos `[l, r]`. Ou seja, todos os elementos entre `l` e `r`, **incluindo** o `r`.

```
int simetrico(int v[], int l, int r) {
    if (l >= r) {
        return 1;
    }
    return v[l] == v[r] && simetrico(v, l + 1, r - 1);
}
```

Exercício: Para quais entradas não precisamos fazer chamada recursiva?

Exercício: O programa termina?

Exercício: O programa funciona, supondo que as chamadas recursivas retornam o valor correto?