

Algoritmos e Programação B

Algoritmos Recursivos

Recursividade ou Recursão

- Recursividade ou recursão é encontrada principalmente na matemática, quando uma função chama a si mesma, ou é usada ela mesma para sua própria definição.
- Um exemplo clássico de problema recursivo é a definição do fatorial de um número (n!):


$$n! = \begin{cases} 1, & \text{se } n=0 \\ n * (n-1)!, & \text{se } n \neq 0 \end{cases}$$

Fatorial

- O cálculo do fatorial de um número pode ser feito de forma iterativa:

```
1  #include <stdio.h>
2
3  unsigned int fatorial(unsigned int n){
4      int i;
5      unsigned int f=1;
6
7      for(i = 1; i <= n; i++){
8          f = f * i;
9      }
10
11     return f;
12 }
13
```

Fatorial

```
13  
14  int main(){  
15     unsigned int d;  
16  
17     scanf("%i", &d);  
18  
19     printf("O fatorial de %i = %i\n", d, fatorial(d));  
20  
21     return 0;  
22 }  
23
```

Algoritmo Recursivo

Consiste em



- diminuir sucessivamente o problema a ser resolvido em um problema menor ou mais simples, até resolver o problema de forma direta.
- pelo menos uma condição de parada (deve estar em pelo menos um local do algoritmo).

Deve possuir



Algoritmos Recursivos

- Podem ser aplicados em:
 - Problemas envolvendo árvores (estruturas de dados);
 - Analisadores léxicos recursivos de compiladores;
 - Problemas que envolvem tentativa e erro (*backtracking*).

Algoritmos Recursivos

Para o desenvolvimento de algoritmos recursivos, são utilizadas funções que fazem uso de recursão ou também chamadas de **funções recursivas**.

Funções Recursivas

- Recursividade consiste em fazer uma função chamar a si mesma direta ou indiretamente. Uma função é recursiva se contém, pelo menos, uma chamada a si própria.
- Para o cálculo do Fatorial, podemos usar a definição matemática:

$$n! = \begin{cases} 1, & \text{se } n=0 \\ n * (n-1)!, & \text{se } n \neq 0 \end{cases}$$

Fatorial

```
1  #include <stdio.h>
2
3  unsigned int fatorial(unsigned int n){
4
5      if (n == 0)
6          return 1;
7      else
8          return n * fatorial(n-1);
9  }
```

Fatorial

```
10
11  ┌ int main(){
12  │   unsigned int d;
13  │
14  │   scanf("%i", &d);
15  │
16  │   printf("O fatorial de %i = %i\n", d, fatorial(d));
17  │
18  │   return 0;
19  │ }
20
```

Outro exemplo: cálculo da potência x^y

- Qual é o algoritmo iterativo para o cálculo da potência?
- A partir da definição matemática recursiva

$$x^y = \begin{cases} 1 & y=0 \\ x * x^{y-1} & y>0 \end{cases}$$

vamos construir a função recursiva.