

# Algoritmos e Estruturas de Dados

## RECURSIVIDADE

Prof. Dr. André Mendes Garcia

# Recursividade

- As funções recursivas são aquelas que fazem chamadas de si próprias
- Em outras palavras, uma função recursiva é chamada dentro dela mesma
- É necessário ter um controle de parada através de uma estrutura condicional para que não entre em uma recursividade infinita
- Internamente, quando uma função faz a chamada de si mesma, o fluxo de execução desta chamada é inserida em uma pilha, e à medida que o algoritmo é executado, esta pilha é esvaziada executando cada linha de execução inserida na pilha
- Vantagens:
  - Complexidade Logarítmica – Melhor desempenho  
**Mas não são todos algoritmos (Fibonacci por exemplo)**
  - Permite implementação fácil em programação paralela

# Recursividade

- Exemplo: Calcular o fatorial do número  $N$

- Maneira **Iterativa**

$$N! = N * (N - 1) * (N - 2) * (N - 3) \dots * 1$$

- Maneira **Recursiva**

$$N! = \begin{cases} N * (N - 1)!, & \text{para } N > 1 \\ 1, & \text{para } N = 1 \end{cases}$$

- Exemplo:  $N = 5$

$$5! = 5 * 4 * 3 * 2 * 1 = 120$$

# Recursividade

- Exemplo: Calcular o fatorial do número  $N$

```
int Fatorial(int n)
{
    if( n > 1 )
    {
        return n * Fatorial(n-1);
    }
    else
    {
        return 1;
    }
} // int Fatorial(int n)
```

# Recursividade

- Exemplo: Sequência de Fibonacci

1, 1, 2, 3, 5, 8, 13, 21, 34 ...

Índice	1	2	3	4	5	6	7	8	9	...
Valor	1	1	2	3	5	8	13	21	34	...

- Fazer uma função recursiva onde o índice da sequência é passado como parâmetro e o valor da sequência de Fibonacci correspondente deve ser retornado
- Exemplo:  $\text{Fibonacci}(8) = 21$

# Recursividade

- Exemplo: Sequência de Fibonacci

```
int Fibonacci(int n)
{
    if( n == 1 || n == 2)
    {
        return 1;
    }
    else
    {
        return Fibonacci(n-1) + Fibonacci(n-2);
    }
} // int Fibonacci(int n)
```

# Recursividade

- Exemplo: Método de Ordenação QuickSort

```
void QuickSort(int *v, int i, int f)
{
    int k;

    if( i < f)
    {
        Particionar(v, i, f, k);

        // Ordenando o Seguimento da Esquerda
        QuickSort(v, i, k-1);

        // Ordenando o Seguimento da Direita
        QuickSort(v, k+1, f);
    }
} // QuickSort
```

# Recursividade

- Exemplo: Busca Binária
  - Consiste em pesquisar um valor dentro de um vetor, analisando sempre a posição central de segmentos do vetor
  - Para utilizar a busca binária, o vetor **DEVE ESTAR ORDENADO**
  - O exemplo a seguir ilustra a busca do valor **258** dentro do vetor **V** de 17 posições, através da busca binária



## Exemplo: Busca Binária

Chave 258

	índice	V
i	0	-98
	1	-8
	2	0,2
	3	2
	4	4
	5	5
	6	7
	7	7,5
	8	47
	9	51
	10	87
	11	258
	12	357
	13	405
	14	487
	15	7682
	j	16

BuscaBin(V, 0, 16, Chave)

## Exemplo: Busca Binária

Chave 258

	índice	V
i	0	-98
	1	-8
	2	0,2
	3	2
	4	4
	5	5
	6	7
	7	7,5
(i+j)/2 = 8		8 47
	9	51
	10	87
	11	258
	12	357
	13	405
	14	487
j	15	7682
	16	98741

BuscaBin(V, 0, 16, Chave)

Chave 258

	índice	V
	0	-98
	1	-8
	2	0,2
	3	2
	4	4
	5	5
	6	7
	7	7,5
	8	47
i	9	51
	10	87
	11	258
	12	357
(i+j)/2 = 13		13 405
	14	487
	15	7682
j	16	98741

BuscaBin(V, 9, 16, Chave)

## Exemplo: Busca Binária

Chave 258

	índice	V
i	0	-98
	1	-8
	2	0,2
	3	2
	4	4
	5	5
	6	7
	7	7,5
$(i+j)/2 = 8$	8	47
	9	51
	10	87
	11	258
	12	357
	13	405
	14	487
	15	7682
j	16	98741

BuscaBin(V, 0, 16, Chave)

Chave 258

	índice	V
	0	-98
	1	-8
	2	0,2
	3	2
	4	4
	5	5
	6	7
	7	7,5
	8	47
i	9	51
	10	87
	11	258
	12	357
$(i+j)/2 = 13$	13	405
	14	487
	15	7682
j	16	98741

BuscaBin(V, 9, 16, Chave)

Chave 258

	índice	V
	0	-98
	1	-8
	2	0,2
	3	2
	4	4
	5	5
	6	7
	7	7,5
	8	47
i	9	51
	10	87
$(i+j)/2 = 11$	11	258
	12	357
j	13	405
	14	487
	15	7682
	16	98741

BuscaBin(V, 9, 12, Chave)

# Recursividade

- Exemplo: Busca Binária – Implementação
  - Fazer uma função com o seguinte protótipo:

$$BuscaBin(V, i, j, Chave) = \begin{cases} -1 & \text{caso não encontre o valor} \\ \text{Ou a posição onde encontrou o valor} \end{cases}$$

- Onde:
  - V : Vetor a ser pesquisado
  - i : Posição inicial do seguimento de busca
  - j : Posição final do seguimento de busca
  - Chave : Valor a ser procurado

# Recursividade

- Exemplo: Busca Binária Implementação

```
//-----  
// Busca Binária  
//-----  
int BuscaBin(float *V, int i, int j, float Chave)  
{  
    int k;  
  
    if( i <= j )  
    {  
        k = (i+j)/2;  
  
        if( V[k] == Chave )  
        {  
            return k;  
        }  
        else  
        {  
            if( Chave < V[k] )  
            {  
                return BuscaBin(V, i, k-1, Chave);  
            }  
            else  
            {  
                return BuscaBin(V, k+1, j, Chave);  
            }  
        }  
    }  
    else  
    {  
        return -1;  
    }  
}  
  
} // BuscaBin
```

# Recursividade

- Exercícios com funções recursivas
  1. Implementar em um só programa as seguintes funções recursivas:
    1. Calcular o Fatorial de um determinado número
    2. Gerar um determinado elemento da sequência de Fibonacci
    3. Cálculo da potência de um número, exemplo:  $\text{Potencia}(2,3) = 8$
    4. Determinar a soma dos  $n$  primeiros números naturais
    5. Determinar a soma dos  $n$  primeiros números pares
  2. Implementar a Busca Binária