

# INE5408 – Estruturas de Dados

Semestre 2008/1  
Prof. Leandro J. Komosinski

## RECURSIVIDADE

# Recursividade na Matemática

É um método de definição de função.

A função que está sendo definida é aplicada dentro da própria função que está sendo definida.

# Recursividade na Matemática

Exemplo clássico: função que calcula o fatorial (!) de um número:

$$0! = 1$$

$$N! = N * (N-1)!$$

# Recursividade na Computação

A recursividade é uma **técnica de programação**.

Se um problema é difícil de resolver então divide-se o problema em subproblemas do mesmo tipo. Esta técnica é chamada de **dividir para conquistar** (*divide and conquer*).

# Recursividade na Computação

Exemplo clássico: calcular  $x^y$  (x elevado à potência y)

A solução tradicional, usada por iniciantes em programação, é multiplicar o número x y vezes usando um para-faça.

# Recursividade na Computação

Exemplo clássico: calcular  $x^y$  (x elevado à potência y)

Usando recursividade podemos reduzir significativamente o número de multiplicações.

Idéia:  $x^y = x^{(y/2)} * x^{(y/2)}$

Exemplo:  $2^6 = 2 * 2 * 2 * 2 * 2 * 2 = (2^3) * (2^3)$

Note que o problema  $2^6$  foi dividido em dois problemas:  $2^3$  e  $2^3$ . Uma vez calculado o valor de  $2^3$  não é necessário calcular novamente.

# Exemplos de Programas Recursivos

```
public class Fatorial {  
    public static long calcule(int n) {  
        long resposta = 1;  
        if (n > 0)  
            resposta = n * calcule(n - 1);  
        return resposta;  
    }  
}
```

# Exemplos de Programas Recursivos

Série de Fibonacci é uma série formada pelos números:

1 1 2 3 5 8 13 21 ...

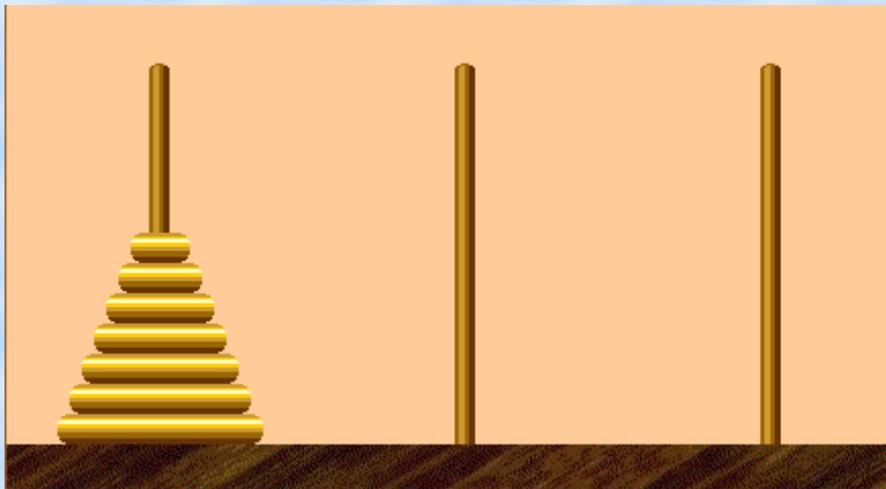
```
public class Fibonacci {  
    public static long calcule(int n) {  
        long resposta = 1;  
        if (n > 2)  
            resposta = calcule(n - 1) + calcule(n - 2);  
        return resposta;  
    }  
}
```

# Exemplos de Programas Recursivos

```
public class FastPower {  
    public static long calcule(int num, int potencia) {  
        long resposta = num;  
        if (potencia > 1) {  
            long numSobre2 = calcule(num, potencia / 2);  
  
            if (potencia % 2 != 0) // potência ímpar  
                resposta = numSobre2 * numSobre2 * num;  
            else  
                resposta = numSobre2 * numSobre2;  
        }  
        return resposta;  
    }  
}
```

# Exemplos de Programas Recursivos

Torre de Hanoi: Mover os  $N$  discos de de um pino até outro pino usando pino auxiliar sendo que somente pode-se mover um disco de cada vez e não é permitido colocar um disco sobre outro disco menor.



# Exemplos de Programas Recursivos

```
public class Hanoi {  
    static int cont = 0;  
    public static void resolve(int qtd, char de, char para, char usando) {  
        System.out.println("Mover " + qtd + " discos do pino " + de +  
            " para o pino " + para + " usando o pino " + usando);  
        mova(qtd, de, para, usando);  
    }  
    ... // continua na próxima transparência
```

# Exemplos de Programas Recursivos

```
private static void move(int qtd, char de, char para, char usando) {  
    if (qtd == 1) {  
        cont++;  
        System.out.println(cont + " Mover disco do pino " +  
            de + " para o pino " + para);  
    } else {  
        move(qtd - 1, de, usando, para);  
        move(1, de, para, usando);  
        move(qtd - 1, usando, para, de);  
    }  
}
```