

# Funções

## Mais alguns exemplos - Repetição de código

```
programa {  
    funcao inicio(){  
        inteiro i  
        escreve_linha()  
        escreva("Numeros entre 1 e 5\n")  
        escreve_linha()  
        para(i=1; i<=5; i++)  
            escreva(i, "\n")  
        escreve_linha()  
    }  
    funcao escreve_linha(){  
        para(i=0; i<20; i++)  
            escreva(" ")  
        escreva("\n")  
    }  
}
```

Observe a diferença ao  
encapsularmos esse código  
repetido em uma função =D



Saída:

\*\*\*\*\*

Numeros entre 1 e 5

\*\*\*\*\*

1  
2  
3  
4  
5

\*\*\*\*\*

# Funções

## Mais alguns exemplos - Repetição de código

```
programa {  
  funcao inicio(){  
    inteiro i  
    escreve_linha()  
    escreva("Numeros entre 1 e 5\n")  
    escreve_linha()  
    para(i=1; i<=5; i++)  
      escreva(i, "\n")  
    escreve_linha()  
  }  
  funcao escreve_linha(){  
    para(i=0; i<20; i++)  
      escreva(" ")  
    escreva("\n")  
  }  
}
```

Observe a diferença ao  
encapsularmos esse código  
repetido em uma função =D



Saída:

\*\*\*\*\*

Numeros entre 1 e 5

\*\*\*\*\*

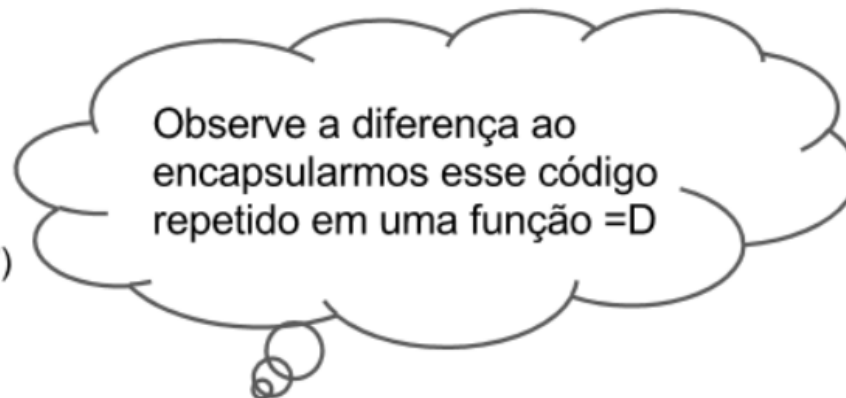
1  
2  
3  
4  
5

\*\*\*\*\*

# Funções

## Mais alguns exemplos - Repetição de código

```
programa {  
    funcao inicio(){  
        inteiro i  
        escreve_linha()  
        escreva("Numeros entre 1 e 5\n")  
        escreve_linha()  
        para(i=1; i<=5; i++)  
            escreva(i, "\n")  
        escreve_linha()  
    }  
    funcao escreve_linha(){  
        para(i=0; i<20; i++)  
            escreva(" *")  
        escreva("\n")  
    }  
}
```



Saída:

\*\*\*\*\*

Numeros entre 1 e 5

\*\*\*\*\*

1  
2  
3  
4  
5

\*\*\*\*\*

Além do método  
`escreve_linha()`, temos  
mais alguma outra função?

# Funções

## Funções de bibliotecas

- Nós vimos várias funções como **escreva()**, **leia()**, **limpa()**.
- Estas funções são métodos padrões já disponíveis em qualquer programa do PortugolStudio. Além dessas funções, podemos adicionar outras funções através da importação de bibliotecas.

**programa**

```
{  
    inclua biblioteca Matematica --> mat  
    funcao inicio()  
    {  
        real numero = 4.0  
        real raiz = mat.raiz(numero, 2.0) // Obtém a raiz quadrada do número  
        escreva("A raiz quadrada de ", numero , " é: ", raiz, "\n")  
    }  
}
```

# Recursividade

Função que chama ela mesma para resolver problemas recursivos.

Mas o que é recursão?

# Recursividade

Função que chama ela mesma para resolver problemas recursivos.

## Mas o que é recursão?

- Em Matemática e Ciência da Computação, uma classe de métodos tem comportamento recursivo quando eles podem ser definidos por duas propriedades:
  - Um caso base simples ( ou vários casos )
  - Um conjunto de regras que reduz todos os outros casos para o caso base
- Exemplo : Fatorial de um número inteiro positivo!!
  - $5! = 5 * 4 * 3 * 2! * 1!$

# Recursividade

Passos para escrever uma função recursiva:

1. Um algoritmo recursivo deve ter um *caso básico*;
2. Um algoritmo recursivo deve mudar o seu estado e se aproximar do caso básico;
3. Um algoritmo recursivo deve chamar a si mesmo, recursivamente.

# Recursividade

## Fatorial Recursivo

```
programa {  
    funcao inteiro fatorial(inteiro n){  
        se(n == 0){  
            retorne 1  
        } senao {  
            retorne n * fatorial( n - 1 )  
        }  
    }  
}
```

### Execução : 4 fatorial

```
n = 4  
retorne 4 * fatorial(3)  
    n = 3  
    retorne 3 * fatorial(2)  
        n = 2  
        retorne 2 * fatorial(1)  
            n = 1  
            retorne 1 * fatorial(0)  
                retorne 1
```



# Retrospectiva

## Como estamos?

- Conteúdo
  - O que é um algoritmo;
  - O que é um programa;
  - Qual ferramenta utilizaremos;
  - Operações de entrada e saída;
  - O que são variáveis e constantes;
  - Desvios condicionais ( se e senão );
  - Operadores lógicos ( E, OU ... );
  - Laços de Repetição;
  - Funções;
  - Uso de Bibliotecas;
  - Recursividade;