


# Funções ou Sub-Algoritmos

- São trecho de algoritmos que efetuam um ou mais tarefas
- Podem funcionar em conjunto com qualquer outro algoritmo para resolver um determinado problema
- Pode ser reutilizado em outros pontos do algoritmo
- Reduzem o tamanho do código
- Facilitam a compreensão e visualização do algoritmo
- A “função” sempre deve informar o tipo do valor que retorna

- A “função” é declarada em qualquer parte do programa
- Sempre retorna apenas um valor ao algoritmo que o chamou


```
public void inicio() {  
    String produto = leTexto("Informe o Nome do Produto");  
    escrevaL("O produto ", produto,  
            " terá um custo de R$", calcula());  
}
```



```
double calcula() {  
    double taxa = 1.15;  
    double valor = leReal("Informe o Preço");  
    return valor * taxa;  
}
```


- Uma “função” pode opcionalmente receber uma lista de argumentos
- Estes argumentos são visíveis somente para a função

```
public void inicio() {  
    String produto = leTexto("Informe o Nome do Produto");  
    double preco = leReal("Informe o Preço");  
    escrevaL("O produto ", produto,  
            " terá um custo de R$", calcula(preco));  
}  
  
double calcula(double valor) {  
    double taxa = 1.15;  
    return valor * taxa;  
}
```



- Quando a “função” não retornar valor o tipo a ser informado deve ser “**void**”

```
public void inicio() {  
    String produto = leTexto("Informe o Nome do Produto");  
    double preco = leReal("Informe o Preço");  
    apresentaCalculo(produto, preco);  
}
```





```
void apresentaCalculo(String nome, double valor) {  
    double taxa = 1.15;  
    double total = valor * taxa;  
    escrevaL("O produto ", nome,  
            " terá um custo de R$", total);  
}
```

- Uma “função” pode requisitar a execução de outra “função”
- Quando esta execução é para a própria “função” que faz a requisição, chamamos esta função de “**função recursiva**”

```
public void inicio() {  
    int num = leInteiro("Informe um nº para o calculo de Fibonacci");  
    escrevaL("O valor calculado com a função fibonnaci para ",  
            num, " é: ", fibonacci(num));  
}
```

 *Requisição Inicial*

```
public int fibonacci(int num) {  
    int fib = 1;  
    if (num > 2)   Requisição Recursiva  
        fib = fibonacci(num - 2) + fibonacci(num - 1);  
  
    return fib;  
}
```

# Fim