

# Funções/procedimentos/métodos

ICC1-Aula 6

# O conceito matemático de função

- uma visão prática (e parcial)...

$$f(x) = ax + b$$

$$f(b) = ax + b$$

$$f(5) = a \cdot 5 + b$$

$$f(5) = a \cdot x + 5$$

Dois “momentos”: definição e uso

$$g(x, y, z) = ax + by + cz$$

$$g(5.5, -3.2, 2.7) = a \cdot 5.5 + b \cdot (-3.2) + c \cdot 2.7$$

Numa função matemática, a avaliação da função não altera nada fora da função.

- Não há construção na linguagem que permita outro comportamento.

$$f(x) = ax + b \quad g(x, y, z) = ax + by + cz$$

$$\left\{ \begin{array}{l} g(5.5, -3.2, 2.7) = a \cdot 5.5 + b \cdot (-3.2) + c \cdot 2.7 \\ e \\ f(x) = bx \end{array} \right.$$

Isto não é uma  
função  
matemática!

# Funções em programação

- Existem linguagens estritamente funcionais, em que as funções de programação também não mudam nada fora dela. Ex. Scheme, Haskell, Erlang... este é o principal comportamento de linguagens funcionais.
- em programação é muito mais comum uma variação chamada “procedimento”, que PODE mudar variáveis fora dele se permitido e desejado – é o que acontece em PASCAL, C, Java,... esta é uma característica de linguagens procedurais.

# Funções em programação - declaração

```
double f(double x) {  
    return a*x+b;  
}  
// falta alguma coisa??
```

# Funções em programação - declaração

```
double f(double x) {  
    double a=...;  
    double b=...;  
    return a*x+b;  
}  
// sim! declarar e inicializar as  
    constantes!
```

# poderíamos fazer assim:

```
double f(double x, double a, double b) {  
    return a*x+b;  
}  
// precisa declarar e inicializar a e b agora?
```

# e como usar???

```
public static void main (String args[]) {  
    double y;  
    y=f(5.45);  // deste jeito!  
    System.out.println (y);  
}
```

```
public static void main (String args[]) {  
    double y;  
    y=f(5.45, 7.4, 8.4);  // deste jeito!  
    System.out.println (y);  
}
```



# Questões que precisam ser resolvidas

- Um comando dentro de uma função pode modificar os valores das variáveis de fora da função??
- Um comando fora de uma função pode modificar os valores das variáveis de dentro da função??
- ... mas de outro jeito pode ser útil...
- SIM!! Dependendo da parte do código em que uma variável é declarada e do modificador de acesso, é possível “mudar a regra”.

# ... e onde dá para declarar variáveis??

```
class Ola {  
    public static void main (String args[]) {  
        System.out.println ("Olá");  
    }  
}
```

```
// usar uma variável antes de declará-la ou  
inicializá-la causa erro.
```

# Blocos de código {} também delimitam escopo.

```
class Ola {  
    public static void main (String args[]) {  
        {  
            int i;  
            i=10;  
        }  
        System.out.println (i);  // erro!  
    }  
}
```

# e quanto a if, while, for??

- para if, while e do..while, se declarar dentro do bloco, a condição está fora do escopo.
- para “switch”, não é possível declarar fora de algum “case”.
- para for, isto funciona:

```
for (int i=0;i<10;i++) {  
    System.out.println (i);  
}
```

# Composição de funções

$$f(w) = aw + b$$

$$g(x, y, z) = ax + by + cz$$

$$x = f(w) \Rightarrow$$

$$g(f(w), y, z) = a \cdot (a \cdot w + b) + by + cz$$

# Para que servem funções??

- Eliminar redundância de código;
- Otimizar uso de memória;
- Evitar uso e reuso inadequado de variáveis;
- Aumentar a clareza do código;

# Provocação

- Como é possível implementar funções/métodos no HIPO?