

Fundamentos da Programação

**O que é
programação?**



Linguagens de programação



Alto Nível

```
1 print("Hello World!")
```



Baixo Nível

```
1 lea si, string
2 call printf
3 hlt
4 string db "Ola mundo!", 0
5 printf PROC
6     mov AL, [SI]
7     cmp AL, 0
8     je pfend
9     mov AH, 0Eh
10    int 10h
11    inc SI
12    jmp printf
13 pfend:
14    ret
15 printf ENDP
```

Variáveis e Operadores

- Tipos de Dados;
- Limitações;
- **Operadores:**
 - Matemáticos (+, -, /, *, módulo)
 - Comparação;
 - Atribuição;
 - Negação;



Estruturas de Condição

- if/else
- switch/case;
- **Operadores Lógicos:**
 - Conjunção (E, AND, &&);
 - Disjunção (OU, OR, ||);
 - Negação (NOT, !)

```
int idade = 20;

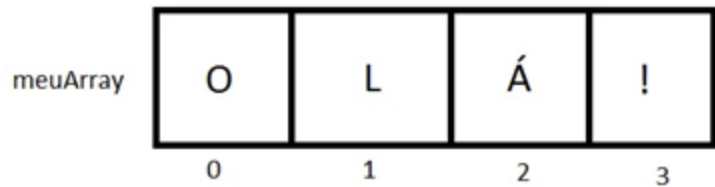
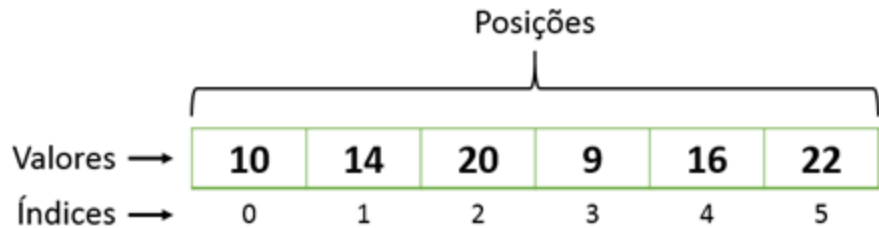
String country = "Brasil";

if(idade < 18) {
    System.out.println(
        "Usuário não pode beber!");
} else if(idade < 21 && country == "EUA" ) {
    System.out.println(
        "Usuário não pode beber!");
} else {
    System.out.println(
        "Usuário pode beber!");
}

switch (country) {
    case "Brasil":
        System.out.println(
            "Usuário é Brasileiro!");
        break;
    case "EUA":
        System.out.println(
            "Usuário é Norte Americano!");
        break;
    default:
        System.out.println(
            "Usuário com nacionalidade "
            + "não identificada!");
        break;
}
```


Arrays ou Vetores

- Conjunto de Valores de mesmo tipo;
- Possui nome e várias posições que podem ser acessadas através de um índice.



Tamanho do array = 4

Estruturas de Repetição

- Execução de instruções que se repetem;
 - For;
 - Do;
 - While;

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

```
do {  
    System.out.println(  
        "Usuário não é Brasileiro!");  
} while(country != "Brasil");
```

```
while(country == "Brasil") {  
    System.out.println(  
        "Usuário é Brasileiro!");  
}
```


Funções

- Blocos de código que permitem estruturar o código;
- Reutilização de código;
- Organizam o código e permitem que um bloco de comandos seja executada em qualquer momento da linha de execução.

```
public static void main(String[] args) {  
  
    int a = 5;  
    int b = 10;  
  
    exibir(a, b);  
  
    a = quadrado(a);  
    b = quadrado(b);  
  
    exibir(a, b);  
  
}  
  
public static void exibir(int a, int b) {  
    System.out.print("A soma de "+a+" e "  
        + "+b+" é "+(a+b));  
}  
  
public static int quadrado(int n) {  
    return n*n;  
}
```

Testes e Debug

- Verificar funcionamento do código;
- Técnicas para identificar erros a serem corrigidos;

1

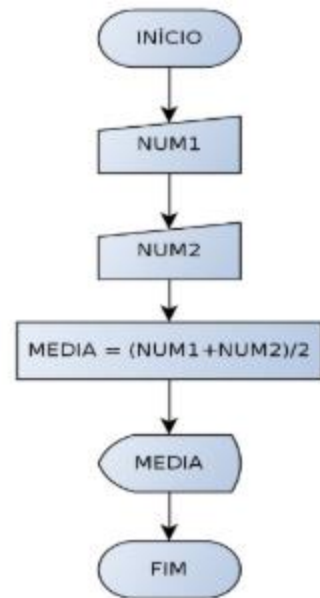
2

3

4

5

6



Instrução	NUM1	NUM2	MEDIA
1	?	?	?
2	5	?	?
3	5	8	?
4	5	8	6.5
5	5	8	6.5
6	5	8	6.5

Exercícios:





Paradigma



Evolução dos Paradigmas



Assembly

LISP, Scheme,
Haskell, Clojure

Algol 68, Cobol,
Linguagem C

C#, Java, Ruby,
Python

Linguagem de Montagem

Programação Funcional

Programação Estruturada

Programação Orientada a
Objetos



Programação Estruturada

- Surgiu no início da década de 60 mediante a *Crise do Software*
- Características: Uso de subrotinas, Laços de repetição, condicionais e estruturas em bloco.
- Foi a base para a Orientação a Objetos



Programação Orientada a Objetos

- Surgiu nos anos 60 através da linguagem Simula
- Alan Kay considerado um dos criadores do termo "Programação Orientada a Objetos"
- Objetivos: Facilitar o desenvolvimento de software e representar o mundo real.





