

# PROGRAMAÇÃO I

---

**Prof. Luiz Albano**

@ luiz.albano@ifsp.edu.br



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SÃO PAULO  
Campus Bragança Paulista

# Agenda

## Tema: Estruturas de Repetição

- While
- Do..while
- For
- Instruções break e continue
- Operadores ++ e –



# Introdução

Praticamente todas as linguagens modernas de programação possuem comandos de iteração (também conhecidos como laços de repetição), que permitem ao programador especificar que uma ou mais ações devem ser repetidas enquanto uma determinada condição for verdadeira.

Exemplo:

**Enquanto houver mais itens em minha lista de compras, comprar o próximo item e riscá-lo da minha lista.**



# Introdução

Até o momento desta disciplina escrevemos uma sequência de instruções a serem executadas. Nas aulas anteriores vimos uma nova forma de escrever programas, em que as instruções podem ou não ser executadas mediante uma ou mais condições predefinidas.

A partir desta aula, veremos novas instruções (estruturas) que permitem a repetição de instruções, sendo elas:

- `while`
- `do...while`
- `for`



# Estrutura while



# Estrutura while

A instrução **while** (também chamada de laço while) executa uma instrução ou bloco de instruções enquanto uma determinada condição for verdadeira.

Sintaxe:

```
while(<condição>) {  
    sequência de comandos  
}
```

**A instrução de um laço também é chamada de CORPO DO LAÇO.**

Em um bloco while, coloca-se entre parênteses a **CONDIÇÃO QUE SE TEM QUE VERIFICAR** para que a instrução ou bloco de instruções seja executado.



# Estrutura while

O funcionamento da estrutura pode ser resumido nos seguintes pontos:

1. A condição é avaliada;
2. Se o resultado da avaliação for **Falso** (0 – zero), o laço termina e o programa continua na instrução imediatamente depois do while;
3. Se o resultado da avaliação for **Verdade** (diferente de zero), é executada a instrução (ou bloco de instruções ali presente) associada ao while;
4. Volta-se ao ponto 1.



# Estrutura while

## Exemplos comentados:

1. Escreva um programa que exiba na tela os primeiros 10 números inteiros:

```
#include <stdio.h>
main() {
    int i = 1;
    while ( i <= 10 ) {
        printf("%d\n", i);
        i = i + 1;
    }
}
```





# Estrutura while

## Exemplos comentados:

### 1. O que faz o programa abaixo?

```
#include <stdio.h>
main() {
    int x = 10;
    while ( x ) {
        printf("%d\n", x);
        x = x - 1;
    }
}
```



# Estrutura while

## Erro comum de programação para a estrutura while:

Fornecer no corpo de uma estrutura while uma ação que posteriormente não torna falsa a condição no teste lógico.

Normalmente, tal estrutura de repetição nunca terminará – este erro é chamado de “**loop infinito**”.



# Instruções break e continue

---





# Instruções break e continue

## **break;**

A instrução break determina o abandono imediato de uma estrutura de repetição ou de seleção (switch).

## **continue;**

A instrução continue funciona como um salto da iteração de uma estrutura de repetição. Ao encontrar uma instrução deste tipo o compilador ignora o restante dos comandos da estrutura de repetição e passa para a próxima iteração (loop).



# Estrutura do...while

---



# Estruturas do...while

A instrução ***do...while*** (também conhecida por laço do...while) difere do laço while porque o teste da condição é realizado no fim do corpo (instrução ou bloco de instruções) do laço e não antes, como acontecia no laço while.

Desta forma o corpo do laço do...while é executado pelo menos uma vez, enquanto no laço while o corpo do laço pode nunca ser executado (caso a condição seja Falsa).

Sua sintaxe é:

```
do {  
    sequência de comandos  
} while (<condição>;
```



# Estruturas do...while

O funcionamento da estrutura do...while pode ser descrito da seguinte forma:

1. A instrução (ou bloco de instruções) é executada;
2. A condição é avaliada;
3. Se o resultado da condição for **Verdade**, volta-se ao ponto 1;
4. Se o resultado da condição for **Falso**, termina o laço e o programa continua na instrução seguinte ao laço.



# Estrutura do...while

**Sempre que for necessário implementar um laço que tenha que executar o seu corpo pelo menos uma vez, utilize o laço do...while. Evite utilizar truques.**

## **Exemplo comentado:**

1. Escreva um programa que exiba na tela um menu com as opções: Cadastrar, Editar, Visualizar e Sair. Ao final o programa deve exibir a opção escolhida, até que este deseje sair.



# Estrutura do...while

## Exemplo comentado (cont.):

```
#include <stdio.h>
main() {
    char opcao;
    do {
        printf("MENU PRINCIPAL\n");
        printf("\tCadastrar\n");
        printf("\tEditar\n");
        printf("\tVisualizar\n");
        printf("\tSairr\n");
        scanf("%c", &opcao);
        fflush(stdin);
        switch ( opcao ) {
            case 'c':
            case 'C': printf("Opcao Cadastrar"); break;
            case 'e':
            case 'E': printf("Opcao Cadastrar"); break;
            case 'v':
            case 'V': printf("Opcao Cadastrar"); break;
            case 's':
            case 'S': printf("Opcao Cadastrar"); break;
            default: printf("Opcao invalida");
        }
    } while ( opcao != 's' && opcao != 'S' );
}
```



# Estrutura for



# Estruturas for

A instrução for (ou laço for) adapta-se particularmente a situações em que o número de iterações é conhecido.

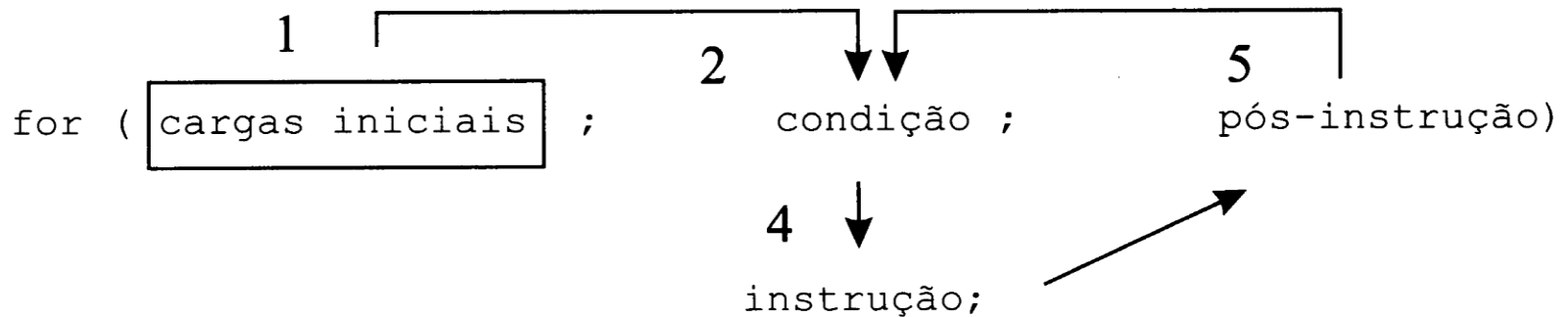
Sua sintaxe é:

```
for (<valores_inicio> ; <condição> ; <pós-instrução>) {  
    sequência de comandos  
}
```

# Estruturas for

Embora apresente um formato estranho, é um laço particularmente bem desenhado, que resume, em uma mesma instrução repetitiva, tudo aquilo de que ela necessita.

Seu funcionamento pode ser resumido pelo seguinte esquema:



# Estruturas for

Funcionamento da estrutura for:

1. O código presente em **cargas iniciais** é executado. Normalmente aqui são iniciadas as variáveis presentes no laço. Este componente do laço for é executado apenas uma única vez;
2. A **condição** é avaliada;
3. Se o resultado da condição retornar o valor **Falso** (zero), então o laço for termina e o programa continua na instrução imediatamente a seguir;
4. Se o resultado da condição retornar o valor **Verdade**, então é executada a **instrução** (ou bloco de instruções) do laço;
5. Depois de executada a instrução presente no laço, é executada a **pós-instrução**. Nesse componente do laço for são normalmente realizadas as alterações necessárias para passar a próxima iteração do laço (incremento ou decremento de variáveis);
6. Voltar ao ponto 2.



# Estrutura for

## Exemplos comentados:

1. Escreva um programa que exiba na tela os primeiros 10 números inteiros:

```
#include <stdio.h>
main() {
    int i;
    for ( i=1 ; i <= 10 ; i=i+1 ) {
        printf("%d\n", i);
    }
}
```

# Estruturas de Repetição - Resumo

	while	for	do...while
Sintaxe	<pre>while(&lt;condição&gt;) {     instruções; }</pre>	<pre>for(&lt;inicialização&gt; ; &lt;condição&gt; ; &lt;incremento&gt;) {     instruções; }</pre>	<pre>do {     instruções; } while(&lt;condição&gt;);</pre>
Executa a instrução	Zero ou mais vezes	Zero ou mais vezes	1 ou mais vezes
Testa a condição	Antes da instrução	Antes da instrução	Depois da instrução
Utilização	Frequente	Frequente	Pouco frequente

# Dúvidas?