



INSTITUTO FEDERAL

Bahia

Campus Santo Antônio de Jesus

<http://www.portal.ifba.edu.br/santoantonio>

LINGUAGEM DE PROGRAMAÇÃO

Prof. George Pacheco Pinto

AGENDA

- ❑ Linguagem C
 - ❑ Estruturas de Repetição
 - ❑ For
 - ❑ While
 - ❑ Do While

ESTRUTURAS DE REPETIÇÃO

- ❑ As vezes é preciso repetir a mesma tarefa para se chegar a um resultado final;
- ❑ Sempre que for necessário repetir determinado trecho do nosso programa utilizaremos as estruturas de repetição/
- ❑ Em C existem 3 opções de repetição:
 - ❑ FOR
 - ❑ WHILE
 - ❑ DO WHILE

ESTRUTURA DE REPETIÇÃO DEFINIDA

- ❑ Estruturas de repetição definidas são facilmente identificadas quando, desde o início da sua execução já está previsto(definido) quantas vezes(loops) ela irá se repetir, ou seja, executar um determinado trecho de código;
- ❑ Cada iteração no processo de repetição é comumente chamada de LOOP;
- ❑ Em C para se utilizar de estruturas de repetição definidas, declaramos blocos do tipo **for**.

FOR

```
for(inicialização; condição; incremento)
{
    //... bloco de código que será repetido
}
```

- ❑ Inicialização: Bloco de código executado no momento da inicialização da estrutura de repetição. Este bloco é executado apenas 1 vez.
- ❑ Condição: Condição que define o momento de saída ou parada da repetição.
- ❑ Incremento: Bloco de código executado após o campo condição de repetição. No primeiro loop este bloco de código não é executado. Comumente utilizado para incrementar a variável que controla a quantidade de loops da estrutura de repetição.

FOR

```
int i;  
printf("Início do Loop");  
for(i = 0; i < 10; i++)  
{  
    printf("Loop: %d", i);  
}  
printf("Fim do Loop");
```

```
Início do Loop  
Loop: 0  
Loop: 1  
Loop: 2  
Loop: 3  
Loop: 4  
Loop: 5  
Loop: 6  
Loop: 7  
Loop: 8  
Loop: 9  
Fim do Loop
```

FOR

- ❑ Cada vez que os comandos são executados, o controle do loop aumenta o valor da variável `i`;
- ❑ O loop será executado enquanto a condição `i < 10` se mantiver verdadeira. Como o valor da mesma no início do loop foi definido como 0 (zero), significa que `i` irá aumentar de 1 em 1 (`i++`) enquanto o seu valor for `< 10`. O que significa dizer que o loop será executado 10 vezes (0 -> 9).

EXEMPLOS

1. Escreva um programa que solicite o nome e ano de nascimento de uma pessoa e imprimir o seu nome e sua idade. O programa deve executar o processo para 50 pessoas.
2. Escreva um programa que leia a data de nascimento (apenas o ano) de 50 pessoas e calcular a média de idade das mesmas.
3. Solicitar do usuário a quantidade de pessoas a serem lidas e para cada pessoa efetuar o ano de nascimento. No final o programa deve calcular a média de idade das mesmas.

ESTRUTURA DE REPETIÇÃO INDEFINIDA

- ❑ O FOR é uma estrutura de repetição definida utilizado para resolver problemas onde se sabe que o número de loops já se encontram subentendidos dentro do enunciado do problema.
- ❑ Existem problemas onde não é possível estabelecer a quantidade de repetições necessárias. Como resolver??

ESTRUTURA DE REPETIÇÃO INDEFINIDA

- ❑ Nas estruturas de repetição indefinidas não sabemos quantas vezes as mesmas serão executadas, daí o termo indefinidas;
- ❑ Neste caso, não se delimita a quantidade de vezes que o comando é executado, na verdade ele será executado indefinidamente enquanto que a condição de repetição estabelecida se encontre verdadeira;
- ❑ Em C para se utilizar de estruturas de repetição indefinidas, declaramos blocos do tipo **while** e **do/while**;

WHILE

```
while(condição de repetição)
{
    //... bloco de código que será repetido
}
```

- ❑ **Condição de Repetição:** Condição que define o momento de saída ou parada da repetição. No momento em que a condição se torna falsa a estrutura de repetição para de executar loops, prosseguindo com o fluxo normal da aplicação.
- ❑ A condição é testada sempre no início da repetição.

EXEMPLO 1

```
#include <stdio.h>
int main()
{
    int numero1, numero2 = 0;
    printf("Informe o primeiro número \n");
    scanf("%d",&numero1);
    while(numero1 != 0){
        printf("Informe o segundo número \n");
        scanf("%d",&numero2);
        printf("Soma dos números: %d \n",numero1 + numero2);
        printf("Informe o primeiro número \n");
        scanf("%d",&numero1);
    }
}
```

DO WHILE

```
do  
{  
//... bloco de código que será repetido  
} while(condição de repetição)
```

- ❑ Condição de Repetição: Condição que define o momento de saída ou parada da repetição. No momento em que a condição se torna falsa a estrutura de repetição para de executar loops, prosseguindo com o fluxo normal da aplicação.
- ❑ A condição é testada sempre no final da repetição.

EXEMPLO 2

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int numero = 0;
```

```
    do{
```

```
        printf("Informe um numero entre 6 e 13 \n");
```

```
        scanf("%d",&numero);
```

```
    } while(numero < 6 || numero >13);
```

```
    printf("Numero digitado:%d \n",numero);
```

```
}
```

REFERÊNCIAS

Consultar ementário.