

---

# Programação Estruturada

## Carolina Aguilar

Repetição – Parte IV

Aula 15 – 2020.1

# Repetição Determinada

---

- Utilizada quando sabemos quantas vezes devemos processar um comando ou um bloco de comandos. Identificamos quando é dito no enunciado quantas vezes processar ou que irá ler quantas vezes processar.

# Repetição Determinada

---

## □ Usando **for**

- A condição é testada antes, ou seja, só executará o bloco de comandos enquanto ela for verdadeira.
- A execução é exatamente igual a execução do **while**.
- Forma geral

```
for (inicializações; condição; atualizações)  
{  
    // bloco de comandos  
}
```

# Repetição Determinada

---

## □ Usando **for**

- *inicializações*: a execução começa pelas inicializações. São executadas uma única vez.
- *condição*: na primeira vez é testada após as *inicializações*. Caso *condição* seja verdadeira, o bloco de comandos será executado. Da segunda vez em diante é testada sempre após a execução das *atualizações*.
- *atualizações*: são sempre executadas após a execução do *bloco de comandos*.

# Repetição Determinada

---

## □ Exemplo 1:

Faça um programa, utilizando a função abaixo, que exiba as tabuadas de 1 a 10.

Faça a função que recebe como parâmetro um número inteiro ( $n$ ) e exiba a tabuada deste número ( $n$ ).

# Repetição Determinada

---

## □ Exemplo 1 - Solução:

```
#include<stdio.h>
void tabuada(int n)
{
    int cont,prod;
    for(cont = 1; cont <= 10; cont++)
    {
        prod = cont * n;
        printf("\n%d x %d = %d", n, cont, prod);
    }
    printf("\n");
}
int main()
{
    int tab;
    for(tab = 1; tab <= 10; tab++)
    {
        tabuada(tab);
    }
    return 0;
}
```

# Repetição Determinada

---

## □ Exemplo 2:

Faça um programa, utilizando a função abaixo, que leia dois números inteiros e exiba os números do intervalo, incluindo os extremos, em ordem crescente. Os números lidos podem não estar em ordem crescente.

Faça a função **intervalo** que receba como parâmetros 2 números inteiros, em ordem crescente, e exiba os números do intervalo, incluindo os extremos.

# Repetição Determinada

---

## □ Exemplo 2 - Solução:

```
#include<stdio.h>
void intervalo(int n1, int n2)
{
    int cont;
    for(cont = n1; cont <= n2; cont++)
    {
        printf("\n%d", cont);
    }
}
int main()
{
    int num1, num2, aux;
    printf("Digite 2 numeros inteiros: ");
    scanf("%d%d", &num1, &num2);
    if (num1 > num2)
    {
        aux = num1;
        num1 = num2;
        num2 = aux;
    }
    intervalo(num1, num2);
    return 0;
}
```



# Exercícios

---

1. Uma loja especializada em microcomputadores possui vários modelos à venda. Cada modelo possui sua própria configuração. O cliente, ao comprar, não pode modificar a configuração dos modelos existentes, mas pode comprar periféricos não oferecidos na configuração por ele escolhida. Para cada venda efetuada pela loja, o cliente escolhe um dos modelos comercializados e o número de periféricos adicionais (0- se não deseja nenhum, ou a quantidade de periféricos desejados). O valor a ser pago pelo computador será o preço do modelo básico escolhido mais o montante pago pelos periféricos. Faça um programa para processar as 100 vendas de uma loja. O programa deverá ler para cada venda: código do modelo escolhido, preço do modelo escolhido, quantidade de periféricos desejados e, para cada periférico o preço do mesmo exibindo o valor da venda. No final, o programa deverá exibir a quantidade de vendas sem periféricos adicionais e o valor médio de todas as vendas. Faça uma função que receba como parâmetro a quantidade de periféricos desejados de uma venda. Esta função deverá ler o preço de cada periférico e retornar o valor a ser pago pelos periféricos.

---

# Fim