

Curso de Tecnologia em Automação Industrial

4º Período

Lógica de programação

Linguagem C

Estruturas de repetição

Até então, os códigos dos programas criados eram executados de forma sequencial (todos os comandos do código são executados), utilizando estruturas de decisão (baseadas em condições ou escolha).

Uma estrutura de repetição serve para fazer com que uma determinada quantidade de instruções ou comandos sejam executados mais de uma vez.

A quantidade de vezes que tais instruções irão repetir dependerá da verificação de uma condição.

Estruturas de repetição

A estrutura irá se repetir, até que uma determinada condição seja satisfeita.

Nessa condição é utilizada uma variável, que nesse caso é chamada de **variável de controle**.

Nas estruturas condicionais, também podemos utilizar as **variáveis acumuladoras**. Esse tipo de variável irá armazenar o resultado acumulado de uma série de valores.

Para isso, o conteúdo das variáveis contadora e acumuladora deverá ser incrementado durante a execução do laço (estrutura de repetição), conforme veremos adiante.

Estruturas de repetição

Inicialização de variáveis.

No caso da utilização de variáveis de controle e acumuladoras, as boas práticas recomendam que seu conteúdo seja inicializado dentro do código fonte, para evitar que algum “lixo” da memória seja acidentalmente atribuído a variável.

Isso será prejudicial ao funcionamento do programa, visto que a variável de controle servirá para controlar o número de vezes que o bloco irá ser executado.

Veremos como isso é feito nos exemplos de estrutura de repetição a seguir.

Estruturas de repetição

As estruturas de repetição, assim como outras estruturas da linguagem C podem ser utilizadas em qualquer momento dentro do código.

Por exemplo, podemos ter um bloco de repetição dentro de uma estrutura condicional If e Else, como também uma estrutura condicional If e Else dentro de uma estrutura de repetição.

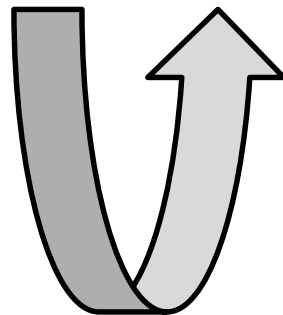
Podemos ainda ter uma estrutura de repetição dentro de outra estrutura de repetição.

Estruturas de repetição - comandos

Do – while

O comando do dará início ao bloco de repetição, o comando while finaliza. Abaixo, sintaxe da utilização dos mesmos:

```
do  
{  
    comando;  
    comando;  
    comando;  
    comando;  
}while condição;
```



Suponhamos que em determinada situação, precisemos repetir uma mensagem por dez vezes na tela. Utilizando estrutura sequencial, o programa ficaria conforme abaixo:

[illegible]

Lógica de programação

Estruturas de repetição - exemplos

Vejamos agora, um programa que faz a mesma coisa, mas usando estrutura de repetição

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c;
    do
    {
        printf("SENAI\n");
        c++;
    }while (c!=10);
    return 0;
    system ("pause");
}
```


Lógica de programação

Estruturas de repetição - exemplos

Vejamos outro exemplo. Faça um programa que escreva os números inteiros positivos menores que 100.

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c=0;
    do
    {
        printf("%d\n",c);
        c++;
    }while (c<100);
}
```

Lógica de programação

Estruturas de repetição - exemplos

Podemos também comparar a variável contadora a outra variável:

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c=0,x;
    printf("Quantas vezes deseja repetir? ");
    scanf("%d",&x);
    do
    {
        printf("SENAI\n");
        c++;
    }while (c!=x);
    return 0;
    system ("pause");
}
```

Estruturas de repetição - exemplos

Vejam os outros exemplos. Faça um programa que faça a soma dos números inteiros positivos menores que 10.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int c=0,t;
    do
    {
        t=t+c;
        c++;
    }while (c!=10);
    printf("%d",t);
}
```

Estruturas de repetição

A estrutura de repetição Do While faz o teste da condição para verificação de repetição dos comandos ou saída da estrutura no final.

A estrutura a seguir faz esse teste no início. O comando utilizado será While, e a estrutura se repetirá enquanto a condição for verdadeira. Abaixo, sintaxe da estrutura:

```
while condição
{
    comando;
    comando;
}
```

Estruturas de repetição

Exemplo (programa que imprime mensagem dez vezes na tela):

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c=0;
    while (c!=10)
    {
        printf("SENAI\n");
        c++;
    }
    return 0;
    system ("pause");
}
```

Estruturas de repetição

Exemplo (programa que mostra números de 0 a 100):

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c=0;
    while (c<100)
    {
        printf("%d\n",c);
        c++;
    }
}
```

Lógica de programação

Estruturas de repetição

Exemplo (programa que mostra números de 0 a 10, comparando a variável contadora a uma outra variável):

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c=0,x;
    printf("Quantas vezes deseja repetir? ");
    scanf("%d",&x);
    while (c!=x)
    {
        printf("SENAI\n");
        c++;
    }
    return 0;
    system ("pause");
}
```

Estruturas de repetição

Estrutura de repetição For.

Na estrutura de repetição For, a variável de controle é inicializada e incrementada no início da estrutura.

Sintaxe:

```
For (valor inicial ; condição final ; incremento)
{
comando;
comando;
comando;
}
```


Estruturas de repetição

Exemplo (programa que imprime mensagem dez vezes na tela):

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c;
    for (c=1;c<=10;c++)
    {
        printf("SENAI\n");
    }
    return 0;
    system ("pause");
}
```

Estruturas de repetição

Exemplo(programa que mostra números de 0 a 100):

```
#include <stdio.h>
#include <cstdlib>
int main()
{
    int c;
    for (c=0;c!=100;c++)
    {
        printf("%d\n",c);
    }
}
```

Estruturas de repetição – exercícios 1

- 1 - Faça um programa que faça a soma dos números pares inteiros positivos menores que 100;
- 2 – Faça um programa que apresente a série de Fibonacci até o décimo termo. A série de Fibonacci é formada pela sequencia: 1, 1, 2, 3, 5, 8, 13, 21, 34 . . .etc, caracterizando-se pela soma de um termo posterior com seu anterior subsequente;
- 3 – Faça um programa que calcule a média entre 10 números, usando no máximo quatro variáveis;
- 4 – Faça um programa que conheça a altura e o sexo de um grupo de alunos (quantidade desconhecida), determine e informe a maior e a menor altura, a média de altura dos homens e das mulheres.
- 5 – A justiça eleitoral encomendou um programa para ser utilizado nas urnas eletrônicas. O programa deverá totalizar os votos de 15 eleitores para os candidatos José da Silva, João dos Santos e Pedro dos Anjos, informando ao final a votação de cada candidato;
- 6 – Faça um programa que conheça o peso e a idade dos animais de uma boiada (quantidade desconhecida) e determine e informe o peso médio, número de bois com idade superior a 36 meses, média de peso dos animais com menos de 12 meses e a somatória do peso dos animais com idade entre 12 e 36 meses.

Estruturas de repetição – exercícios

- 7 – Faça um programa que solicite um número, determine e informe se o mesmo é ou não é um número primo;
- 8 - Faça um programa que calcule o valor da área de uma circunferência ($A = 3,14 \times R^2$) e utilize estruturas de repetição para validar o valor informado para o raio, para que não seja nem menor, nem igual a 0 (zero);
- 9 – Ajuste o programa número 8, de estruturas condicionais, para validar a digitação do salário do indivíduo e não permitir valor informado seja igual ou menor a 0 (zero);
- 10 – Faça um programa que calcule a média entre 10 números, usando no máximo quatro variáveis. Utilize estruturas de repetição para validar e não permitir que sejam informados valores menores ou iguais a 0 (zero);
- 11 – Faça um programa que conheça a altura e o sexo de um grupo de alunos (quantidade desconhecida), determine e informe a maior e a menor altura, a média de altura dos homens e das mulheres. Utilize estruturas de repetição para validar a entrada de dados e não permitir que sejam informados valores diferentes de M ou F para o sexo, e menores ou iguais a 0 (zero) para a altura;
- 12 – A sequencia a seguir está disposta de maneira lógica. Faça um programa que escreva a mesma na tela: 1 1 2 3 3 5 4 7 5 9 6 11 7 13 . . .

Estruturas de repetição – Validação de entrada de dados

O processo de validação é importante na programação, pois permite garantir a integridade dos dados informados ao programa, garantindo que a informação de saída seja correta e satisfatória.

Vamos utilizar como exemplo o programa a seguir: Esse é um fragmento de código de um programa que solicita que uma nota seja informada.

Considerando que as notas informadas devem ser valores entre 0 e 10, não podem ser aceitos valores fora dessa faixa.

Estruturas de repetição - exemplos

Aqui o código fonte do programa, onde não é feito nenhum tipo de validação sobre o dado informado.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
int main()
{
    setlocale(LC_ALL, "");
    float n1;
    printf("Informe uma nota: ");
    scanf("%f", &n1);
    while ((n1 < 0) || (n1 > 10))
    {
        printf("Nota inválida, informe valores entre 0 e 10. ");
        scanf("%f", &n1);
    }
}
```

Lógica de programação

Estruturas de repetição – exercícios 2

Utilize estruturas de repetição para validar as entradas feitas para os programas que utilizam estruturas de múltipla escolha, fazendo com que além de ser mostrada a mensagem de entrada inválida ao ser informado um valor fora da faixa especificada dentro do case, que o programa repita a solicitação de entrada dos dados, até que seja feita a entrada correta.