

LAÇOS DE REPETIÇÃO

Bibliografia:

Cap 5 - Estudo Dirigido de Linguagem C

José Augusto Manzano, Ed. Erica

Aula 5

Objetivos

- Entender laços de repetição
- Aprender como funciona o comando while

while

Sintaxe

```
while <(condição)>
{
    <instruções para a condição verdadeira>;
}
```

A condição é testada antes de se entrar no bloco de comandos. Caso seja falsa, o programa desviará para a instrução logo após o bloco. Se for verdadeira, executará as instruções do bloco, até que ela seja falsa. Cuidado, isto pode gerar um loop infinito.

Exemplo

Programa 3.1.while-exemplo

Este programa pede ao usuário que digite um número de 1 a 10.
Ao final, exibe a taboada do número informado.

Atenção para a instrução `I++`;

Ela é equivalente a `I = I + 1`;

Suponha que `I` contenha o valor 5, considere:

`Z = I++`; // `Z` irá conter 5, e `I` terá o valor alterado para 6

`Z = ++I`; // primeiro é feita a soma, então `I` passa para 6, e depois este valor é transferido para `Z`, que também fica com 6.

O mesmo raciocínio vale para `--` se `I` contém 5

`Z = I--`; // `Z` irá conter 5, e `I` terá o valor alterado para 4

`Z = --I`; // primeiro é feita a subtração, então `I` e `Z` contém 4

Atividade

- Salvar o programa 3.1.while-exemplo como 3.1b.while-decrescente
- Ao invés de mostrar a tabula multiplicando de 1 a 10, exibir ao contrário, multiplicando de 10 até 1. Ex:

$$4 \times 10 = 40$$

$$4 \times 9 = 36$$

... e assim por diante

Dicas de Programação

- 1) Edentar o programa: ou seja, fazer as tabulações de forma hierárquica, com 3 ou 4 espaços, cada vez que sobe de nível.

Evite por exemplo:

```
{  
if (a==0)  
{  
comandos  
}  
else  
{  
comandos  
}  
}
```

Procure digitar desta forma:

```
{  
    if (a==0)  
    {  
        // dê tres ou quatro espaços para os codigos dentro do if  
        comandos  
    } // feche o abre parenteses no mesmo nível hierárquico  
    else // else alinhado com o if correspondente  
    {  
        comandos  
    }  
}
```

// fica claro aqui, a qual abre chaves este fecha chaves se refere

Dicas de Programação

- 1) Não tente digitar todo o programa para só depois compilar. Vá digitando partes pequenas, compilando, e testando gradativamente.
- 2) Ao se deparar com resultados inesperados, crie a prática de exibir mensagens e dar pausa, para ver se o programa passou por onde você queria. Ex:

```
{  
    printf ("teste para ver se passou no ponto 1");  
    system("pause");  
    for (j=0; J<=3; J++)  
    {  
        printf ("teste valor de j = %d \n", j);  
        system("pause");  
        // comandos  
    }  
    printf ("teste para ver se passou no ponto 3");  
    system("pause");
```

Os comandos em azul não faziam parte do programa original. Eles foram inseridos apenas para que você acompanhe a execução e descubra onde está o problema no seu programa.

Exercício

Programa 3.2.TipoTrianguloLoop

Tome como base o programa 2.6.TipoTriangulo

Altere-o para que seja executado tantas vezes quanto o usuário desejar. Ou seja, **crie um loop** utilizando while, e pergunte ao final: “Deseja continuar? [1] Sim / [2] Não”. Se o usuário digitar sim, volta para o início do programa, e pede para o usuário digitar novamente os lados do triangulo. Se digitar 2, sai do loop.

Exercício

Programa: 3.3.Fatorial

Crie um loop, para executar o programa enquanto o usuário responder que deseja continuar.

Solicite um número para calcular a fatorial.

Crie um loop para calcular a fatorial e exiba o resultado.

No loop que calcula a fatorial, você provavelmente irá incrementar um contador *I*, e usar um comando semelhante a

```
FATORIAL = FATORIAL * I;
```

O comando acima pode ser substituído por

```
FATORIAL *= I;
```

Exercício extra

- Elaborar um programa que peça para o usuário digitar um número, e verifique se o número digitado é primo ou não.

Importante

- Tente em casa fazer o programa da Taboada sozinho, você mesmo, sem consultar o programa original.
- Use a Internet para pesquisar outros programas e instruções para ajudá-lo com a sintaxe

Aula 6

Objetivos

- Fazer exercício sobre while para consolidar aprendizado
- Aprender como funciona o comando do... While
- Aprender como funciona o comando for ...

do ... while

Sintaxe

```
do  
{  
    <instruções de comando>  
} while <(condição)>;
```

- Neste caso, o teste lógico é feito somente ao final do bloco. Ou seja, independente se o teste é verdadeiro ou falso, o bloco de comandos é executado.

Exercicio

Programa 3.4.TaboadaDoWhile

Altere o programa 3.1.While-Exemplo para usar o comando do... while, ao invés do comando while.

Programa 3.5.SwitchNumericoLoop

Tomando como base o programa 2.3.b.SwitchNumerico, acrescente ao menu uma nova opção [9] Sair.

O sistema deverá ficar em um loop, e sair apenas se a opção escolhida for 9

for ...

Sintaxe

```
for (<início>; <fim>; <incremento>)
{
    <instruções>;
    <instruções>;
    <instruções>;
}
```

Digite e execute o programa exemplo abaixo:

```
/* contagem crescente */
#include <stdio.h>
#include <stdlib.h>
int main (void)
{
    int l;
    for (l = 1; l <= 10; l ++ )
        printf ("l = %2d\n", l);
    // quando existe apenas um comando, não é preciso usar chaves
    system("pause");
    return 0;
}
```

3.6.a.ForContagemCrescente

Atenção, lembrar que `l++` é a mesma coisa que `l = l + 1`, que poderia também ser utilizado dentro do for. Faça o teste!

Exercícios

Programa 3.6.b.ForContagemDecrescente

Elabore um programa que faça a contagem decrescente de 50 até 40, e exiba os números na tela, de forma semelhante ao programa anterior.

Programa 3.7.ForNumerosPares

Elabore um programa que exiba os número pares de 0 a 100. Utilize o comando for para incremento.

Exemplo 2 contadores

Programa 3.8.ForDoisContadores

Abra o programa, mas não o execute.

Sem executar, escreva em um papel, qual será o resultado exibido pelo programa.

Depois, execute o programa e verifique o resultado.

Exercício

Programa 3.9.Fatorial

Elabore um programa que peça ao usuário um número, e exiba a fatorial dele, utilizando o comando for.

Após exibir o resultado, o programa deve perguntar se o usuário deseja continuar no programa ou sair.

Exercícios extras

1) elabore um programa que peça ao usuário um número. Entre 1 e o número informado, exiba os números divisíveis por 4. (existe um comando que exibe o resto de uma divisão. Caso não se lembre, procure na internet).

2) Peça para o usuário entrar com a idade, e com a quantia que ele consegue depositar na poupança todo mês. Vamos assumir que ele consiga economizar esta mesma quantia todos os meses, até se aposentar aos 65 anos. Para simplificar, vamos dizer que $(65 - \text{idade informada}) * 12 = \text{número de meses para investimento}$, e que o valor que ele irá depositar também será o mesmo todos os meses. Considere que a poupança rende 0,6% ao mês. Calcule quanto esta pessoa terá acumulado de poupança quando ela se aposentar.