

Algoritmos e Programação Estruturada

Estruturas de repetição condicional

Você sabia que seu material didático é interativo e multimídia? Isso significa que você pode interagir com o conteúdo de diversas formas, a qualquer hora e lugar. Na versão impressa, porém, alguns conteúdos interativos ficam desabilitados. Por essa razão, fique atento: sempre que possível, opte pela versão digital. Bons estudos!

Nesta webaula vamos ver sobre as estruturas de repetição condicional.

Estrutura de repetição condicional `while`, `do/while`

Assim como nas estruturas de decisão, as estruturas de repetição desenvolvem aplicações para a otimização do pensamento computacional e ao mesmo tempo, a agilidade nas soluções dos problemas de repetição.

Estrutura de repetição com teste no início – `while`

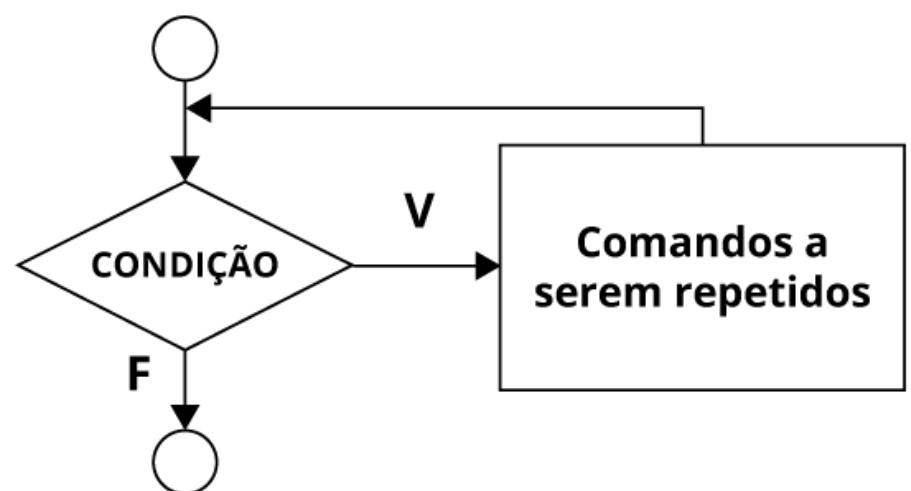
Neste caso algo será repetidamente executado enquanto uma condição verdadeira for verificada, somente após a sua negativa essa condição será interrompida.

Na realização dessa condição, o comando iterativo "`while`", que significa "enquanto" em português realiza o teste no início, antes de executar as ações programadas.

Sintaxe do comando "`while`":

```
while (<condição>)  
{  
  Comando 1;  
  Comando 2;  
  Comando n;  
}
```

Fluxograma – repetição com teste no início



Fonte: elaborado pelo autor.

Quando utilizamos teste no início, pode ocorrer o famoso ***loop*** (laço) **infinito**, que é quando um processo é executado repetidamente. Para que isso não aconteça, você poderá utilizar os seguintes recursos:

Contador

É utilizado para controlar as repetições, quando esta é determinada.

Incremento e decremento

Trabalham o número do contador, seja ele, aumentando ou diminuindo.

Acumulador

Irà somar as entradas de dados de cada iteração da repetição, gerando um somatório a ser utilizado quando da saída da repetição.

Condição de parada

Utilizada para determinar o momento de parar quando não se tem um valor exato desta repetição.

No exemplo de repetição condicional a seguir, utilizando o comando `while` com teste no início, mostrará a palavra “PROGRAMA” dez vezes.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  main()
4  {
5      int cont=0; // foi definido na declaração da variável um valor inicial de "0"
6      while (cont < 10) // Será executado enquanto a cont for menor que 10
7      {
8          printf("PROGRAMA \n");
9          cont++; // será necessário incrementar um valor, para dar sequência no programa
10     }
11     system("PAUSE");
12     return 0;
13 }
```

Estrutura de Repetição condicional com teste no final – do/while

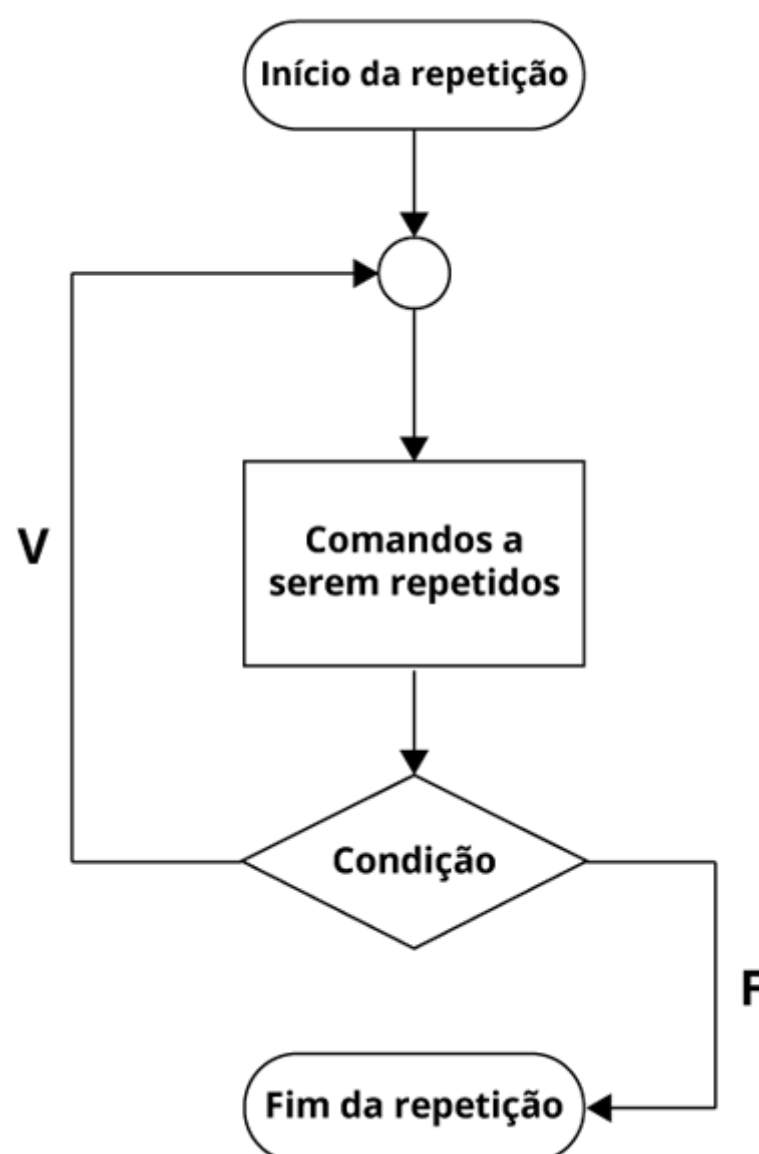
O laço “do-while” analisa a condição ao final do laço, ou seja, os comandos são executados antes do teste de condição.

Neste caso, em específico, o usuário tem a possibilidade de digitar novamente uma nova informação (SCHILDT,1997).

Sintaxe do comando “do/while” :

```
Do
{
comandos;
}
while (condição);
```

Fluxograma – repetição com teste no final



Fonte: elaborado pelo autor.

O exemplo a seguir, realiza um programa que calcula a metragem quadrada de um terreno usando o teste no final para criar a opção de digitar novos valores sem sair do programa.

```

1  #include <stdio.h>
2  main() {
3  float metragem1,metragem2,resultado;
4  int resp;
5  metragem1 = 0;
6  metragem2 = 0;
7  resultado = 0;
8  do
9  {
10     printf("C A L C U L O   D E   M E T R O S   Q U A D R A D O S");
11     printf("\n \n Digite a primeira metragem do terreno: \n");
12     scanf("%f",&metragem1);
13     printf("\n Digite a segunda metragem do terreno: \n");
14     scanf("%f",&metragem2);
15     resultado = (metragem1 * metragem2);
16     printf("\n \n O Terreno tem = %.2f M2 \n",resultado);
17     printf("Digite 1 para continuar ou 2 para sair\n");
18     scanf("%d", &resp);
19 }while (resp==1);
20 return 0;
21 }

```

O exemplo a seguir, realiza um programa que simula uma conta bancária (com tela de opções das transações. Ele repete uma entrada de dados até que determinada condição de saída seja atingida e, em seguida, acumule os valores digitados. Observe que foi utilizado o laço `do-while` para implementar o menu do programa, uma estrutura de repetição usando comparativo. Adaptado do livro do Soffner (2013).

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  main()
4  {
5  float soma=0;
6  float valor;
7  int opcao;
8  do {
9  printf("\n Digite uma Operacao");
10 printf("\n 1. Deposito");
11 printf("\n 2. Saque");
12 printf("\n 3. Saldo");
13 printf("\n 4. Sair");
14 printf("\n Opcao? ");
15 scanf("%d", &opcao);
16 switch(opcao) {
17     case 1: printf("\n Valor do deposito? ");
18             scanf("%f", &valor);
19             soma=soma+valor;
20             break;
21     case 2: printf("\n Valor do saque? ");
22             scanf("%f", &valor);
23             soma=soma-valor;
24             break;
25     case 3: printf("\n Saldo atual = R$ %.2f \n", soma);
26             break;
27     default: if(opcao!=4)
28 printf("\n Opcao Invalida! \n");
29 }
30 }
31 while (opcao!=4);
32     printf("Fim das operacoes. \n\n");
33     system("pause");
34     return 0;
35 }

```

Pesquise mais!

O comando do-while pode ter várias aplicações. Veja o vídeo no YouTube “feito de aluno para aluno” sobre esse tema:

DE ALUNO PARA ALUNO. Programar em C - Como Utilizar "do while" - Aula 13. 24 out. 2012.