

MC102 - Algoritmos e Progração de Computador

Prof. Alexandre Xavier Falcão

6º Aula: Comandos de repetição.

Em muitas tarefas de programação, desejamos que um bloco de comandos seja executado repetidamente até que determinada condição seja satisfeita.

1 Comando do while

O comando **do while** é uma instrução de repetição, onde a condição de interrupção é testada após executar o comando.

```
do {  
bloco de comandos  
} while (expressão lógica);
```

O bloco de comandos é repetido até que a expressão seja falsa. Suponha, por exemplo, um programa para dizer se um número inteiro lido da entrada padrão é par ou ímpar. Desejamos que o programa execute até digitarmos um número negativo.

```
#include <stdio.h>  
  
int main()  
{  
    int n;  
  
    do {  
  
        printf("Digite um número inteiro:");  
        scanf("%d",&n);  
  
        if ((n%2)==0)  
            printf("O número é par\n");  
        else  
            printf("O número é ímpar\n");  
  
    } while (n >= 0);  
  
    return(0);  
}
```

Um exemplo prático interessante é o algoritmo de Euclides para calcular o Máximo Divisor Comum (MDC) de dois números inteiros positivos.

1. Leia m e n .
2. Faça $x \leftarrow m$ e $y \leftarrow n$.
3. Atribua a r o resto da divisão de x por y .
4. Faça $x \leftarrow y$ e $y \leftarrow r$.
5. Volte para a linha 3 enquanto $r \neq 0$.
6. Diga que x é o MDC de m e n .

Codifique este algoritmo em C.

2 Comando while

O comando **while** é uma instrução de repetição, onde a expressão lógica é testada antes de executar o comando. Sua estrutura básica envolve quatro etapas: inicialização de uma variável de controle, teste de interrupção envolvendo a variável de controle, execução do bloco de comandos, e atualização da variável de controle.

```
inicialização
while(expressão lógica)
{
    bloco de comandos
    atualização
}
```

Suponha, por exemplo, um programa que soma n valores reais lidos da entrada padrão e apresenta o resultado na saída padrão.

```
#include <stdio.h>

int main()
{
    int i,n; /* variável de controle i */
    float soma,num;

    printf("Entre com a quantidade de números a serem somados: "); scanf("%d",&n);
    /* inicialização */
    i = 1; soma = 0.0;
    while (i <= n) { /* expressão */
        /* bloco de comandos */
        printf("Digite o %do. número:",i);
        scanf("%f",&num);
```

```

    soma = soma + num;
    i     = i + 1; /* atualização */
}
printf("O resultado da soma é %f\n",soma);

return(0);
}

```

Modifique o programa acima para ele calcular o produto dos n números.

Observe que nos exemplos envolvendo o comando do while, a variável de controle é inicializada e atualizada no bloco de comandos. Neste caso, a construção com do while fica mais elegante do que com o comando while. Por outro lado, em situações onde a variável de controle não participa do bloco principal de comandos, a construção fica mais elegante com o comando while.

Podemos combinar vários comandos if, do while e while aninhados.

```

#include <stdio.h>

int main()
{
    int i,n;
    float soma,num;
    char opt;

    do {
        printf("Digite a opção desejada:\n");
        printf("s para somar números, \n");
        printf("ou qualquer outra letra para sair do programa.\n");
        scanf(" %c",&opt);
        if (opt == 's'){
            printf("Entre com a quantidade de números a serem somados\n");
            scanf("%d",&n);
            i = 1; soma = 0.0;
            while (i <= n) {
printf("Digite o %do. número:",i); scanf("%f",&num); soma = soma + num;
i     = i + 1;
        }
            printf("O resultado da soma é %f\n",soma);
        }
    } while (opt == 's');

    return(0);
}

```

Podemos também criar um laço infinito, substituindo **do while** por **while(1)**, e sair dele com o comando **break** no **else** do **if**. Outro comando de desvio interessante é o comando **continue**. Ele permite desconsiderar o restante do bloco de comandos, voltando para o teste da expressão lógica. Por exemplo, podemos desprezar o processamento de números negativos, acrescentando:

```
if (num < 0) continue;
```

após a leitura do número.

3 Exercício

Altere o programa que calcula o MDC de dois números para usar o comando while, e faça com que ele se repita até digitarmos algo diferente de 'm'.