

Engenharia de Computação

Fundamentos de Programação

Aula 08 – Estrutura de Repetição While

Prof. Muriel de Souza Godoi
muriel@utfpr.edu.br

Laço de Repetição

- É necessário quando se tem que repetir uma determinada instrução, que represente um comportamento padrão, várias vezes.
- Com isso, escreve-se o comportamento padrão uma única vez e ela pode ser executada várias vezes.
- As repetições podem ser controladas por uma condição.
- Deve-se garantir que o laço acabe.
 - Caso contrário entra em "loop", ou seja, repete infinitamente.

Laço de Repetição

- **Exemplo de problema 1:** Faça um programa que calcule a soma de todos os números inteiros compreendidos entre 1 e 5.
- **Exemplo de problema 2:** Faça um programa que calcule a soma de todos os números inteiros compreendidos entre 1 e 1000.

Laço while

- A estrutura **while** verifica no começo do laço de repetição se uma condição é verdadeira ou não. Caso seja verdadeira, o bloco no interior do laço é executado, e assim sucessivamente até que a condição seja falsa.
- O **while** pode ser pensado como *“enquanto a condição for verdadeira, execute o bloco do laço, caso contrário, pare o laço”*

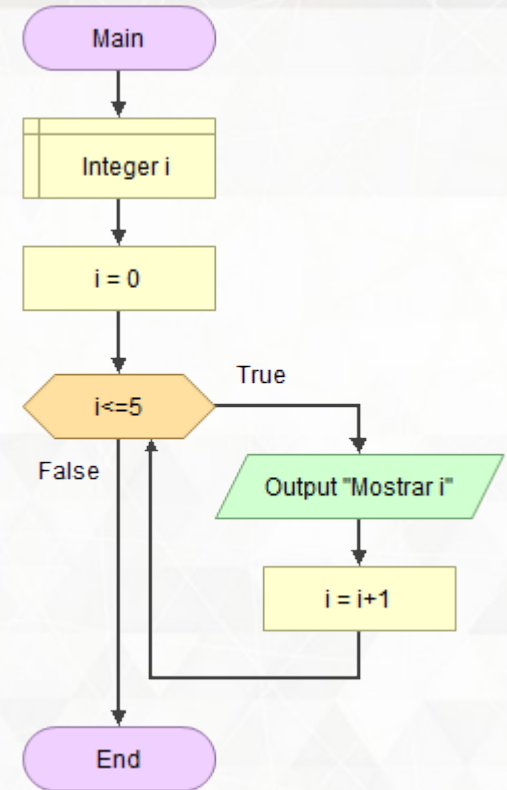
Laço while

Sintaxe:

```
while (/*condição*/) {  
    //instruções que  
    //devem ser repetidas  
} //while
```

Exemplo (incremento):

```
i = 0; //inicializa  
while (i <= 5) {  
    printf("Valor i= %i", i);  
    i++; //atualiza o valor  
} //while
```



Para criar a condição do laço **while** utilize os mesmos operadores relacionais que são utilizados no **if**

Laço while

- Algumas vezes não sabemos quantas vezes o laço irá repetir a execução do bloco.
 - Exemplo de um programa que leia e mostre as teclas digitadas pelo usuário até que digite a tecla 'z'

Exemplo:

```
char tecla;  
tecla='1';  
printf("Mostra teclas ate que digite z... \n");  
while (tecla != 'z') {  
    scanf("%c",&tecla);  
    setbuf(stdin,NULL);  
    printf("Tecla: %c \n", tecla);  
} //while
```

Variável Contadora

- A cada passo a variável conta numero de ocorrências de uma condição
 - Deve ser inicializada e não interferir no laço

Exemplo:

```
int i = 0;
int contapar = 0;
while (i <= 20){
    if (i % 2 == 0){
        contapar++;
    } //if
    i++;
} //while
printf("Números pares = %i\n", contapar);
```

Variável Acumuladora

- A cada passo a variável acumula o resultado parcial
 - Deve ser inicializada e não interferir no laço

Exemplo:

```
int somapar = 0;
int i = 0;
while (i <= 20){
    if (i % 2 == 0){
        somapar = somapar + i;
    }//if
    i++;
}//while
printf("A soma dos pares e %i", somapar);
```


Exercícios com while

- 1) Faça um algoritmo que exiba todos números de 1 a 100.
- 2) Faça um algoritmo que exiba todos números de 100 a 1.
- 3) Crie um algoritmo que exiba todos os números múltiplos de 3 no intervalo de 1 a 322.
- 4) Crie um algoritmo que exiba todos os números pares entre 240 e 730 inclusive.

Exercícios com while

- 5) Faça um programa que receba um número real, calcule e exiba o quadrado dele. O programa deverá repetir esse procedimento para 10 números, um de cada vez.
- 6) Faça um programa que receba um número real, calcule e exiba o quadrado dele. O programa deverá calcular o quadrado de vários números e finalizar quando for digitado um número negativo ou zero.
- 7) Faça um programa que calcule o fatorial de um número a ser digitado.

Exercícios

8) Faça um menu que mostre 3 opções

Menu do programa:

1 - Teste 1

2 - Teste 2

3 – Sair do programa

Digite sua opção:

- A primeira opção imprime o texto “Teste1” e mostra o menu novamente.
- A segunda opção imprime “Teste2” e mostra o menu novamente.
- A terceira opção finaliza o programa.
- Caso digite uma opção diferente, deve mostrar uma mensagem de erro e mostrar o menu novamente.

Exercícios

- 9) Faça um programa que simule um jogo, onde o usuário deverá descobrir um número aleatório escolhido pelo computador (de 1 a 100).
- O usuário poderá realizar até 6 tentativas, o programa deverá retornar as mensagens “muito alto”, “muito baixo” até o usuário acertar o número ou esgotar o número máximo de tentativas



Pesquise sobre como gerar um numero aleatório utilizando a função `rand` da biblioteca `stdlib.h`