

Algoritmo e Programação

Estruturas de Controle e Repetição

Prof. Dr. Gilberto Fernandes Jr.





Estruturas de repetição

- Repetição com teste no início: **while**;
- Repetição com teste no fim: **do ... while**;

Como você faria a tabuada do 2?

```
res = 2*1;
printf("%d \n", res);
res = 2*2;
printf("%d \n", res);
res = 2*3;
printf("%d \n", res);
res = 2*4;
printf("%d \n", res);
...
res = 2*10;
printf("%d \n", res);
```

- E se eu quiser calcular a tabuada dos 10 primeiros termos?
- Temos instruções **repetidas!**
- Então podemos **automatizar!**
- O que se **repete**? E o que está **variando**?

```
.....
.....
.....
```

```
res = 2 * x;
printf("%d \n", res);
x = 1, 2, ..., 10, ... n
```

Laços de Repetição

- Algo será **repetidamente** executado enquanto uma condição verdadeira for atendida
- Essa repetição será interrompida somente quando a condição não for mais satisfeita.
- Os laços precisam de uma **condição de parada** para prevenir o que chamamos de **loop infinito!**

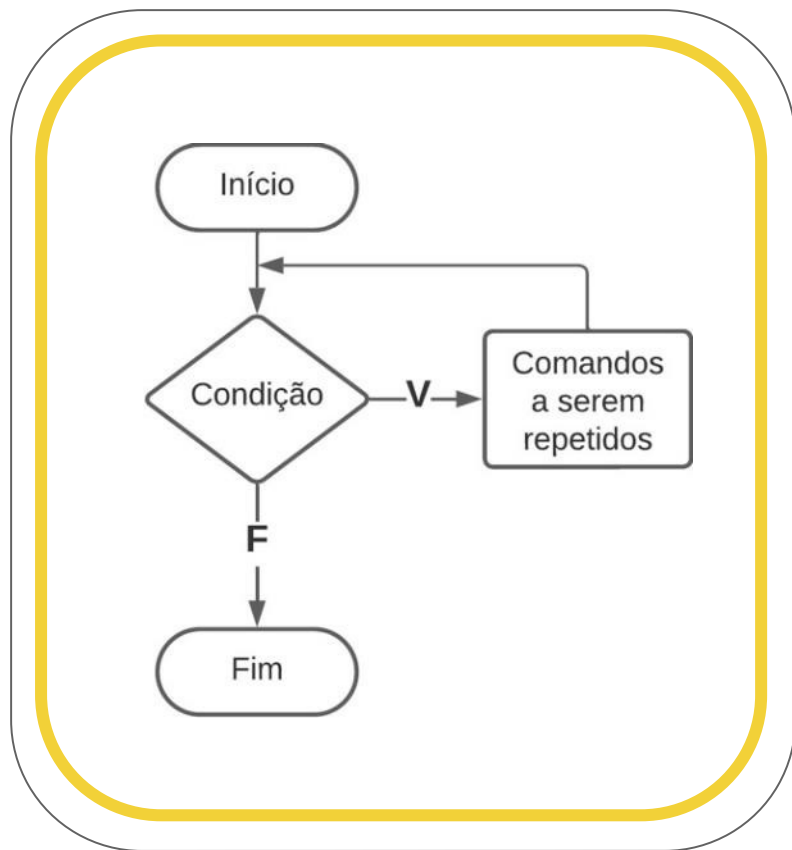
contador / acumulador / incremento / decremento



Repetição com teste no início: **while**

- Uma estrutura de repetição com teste no início não iniciará nenhuma repetição (e os comandos programadas dentro dela) sem primeiro verificar uma condição.





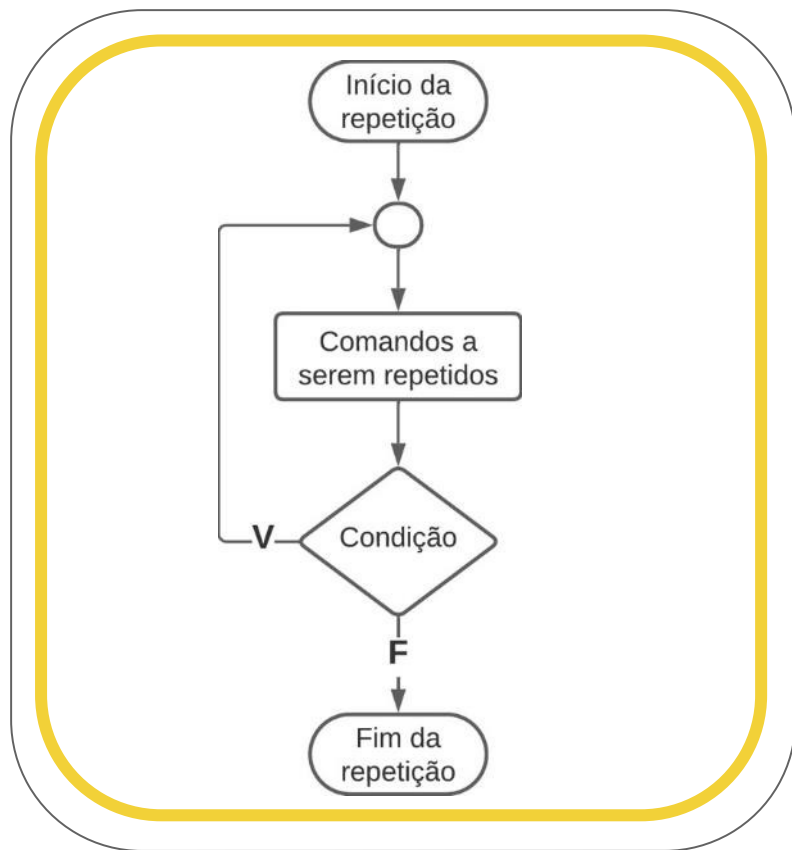
Estrutura **while**

```
while (condição)
{
    // comando 1
    // comando 2
    ...
}
```

Repetição com teste no fim: **do-while**

- Avalia a condição ao **final** do ciclo
- Os comandos são executados antes de verificar a condição, ou seja, o bloco de comandos do laço **sempre é executado ao menos uma vez**





Estrutura **do-while**

do {

// comando 1

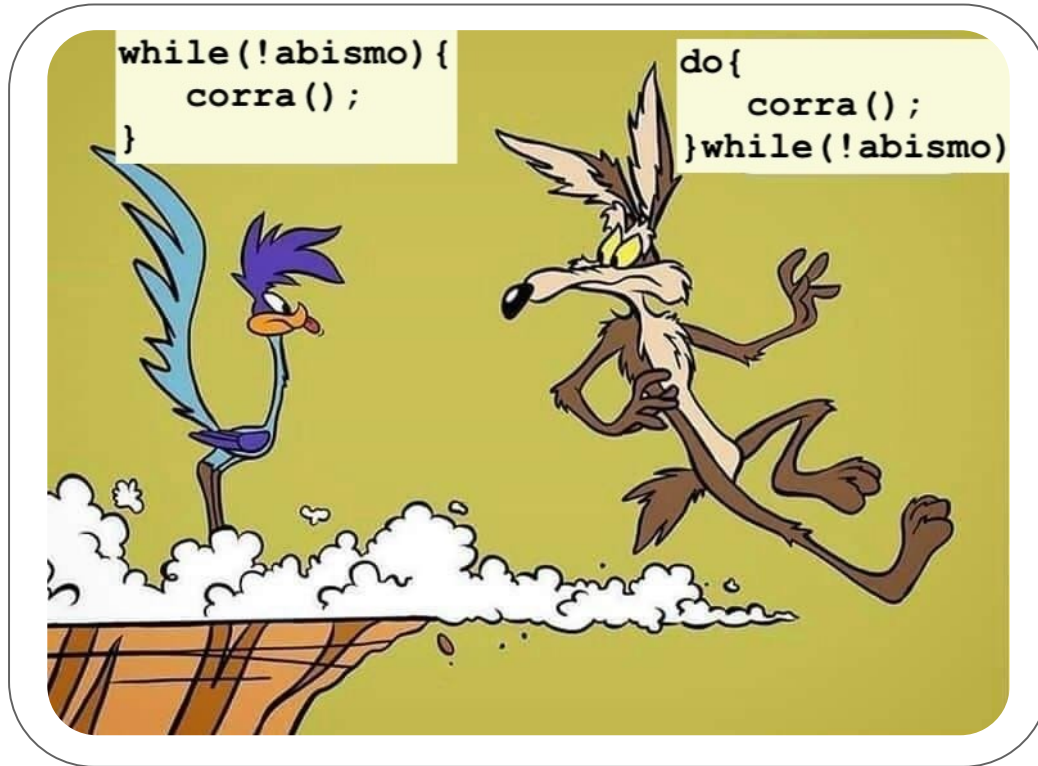
// comando 2

...

} while (condição);

Cuidado! Não se esqueça deste ponto e vírgula (;)

Resumindo...





Vamos Praticar?

Vamos utilizar o Visual Studio Code para construir alguns algoritmos com estruturas de repetição na linguagem C