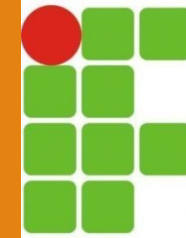


**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
BAHIA

LINGUAGEM DE PROGRAMAÇÃO

COMANDO DE REPETIÇÃO EM C



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA

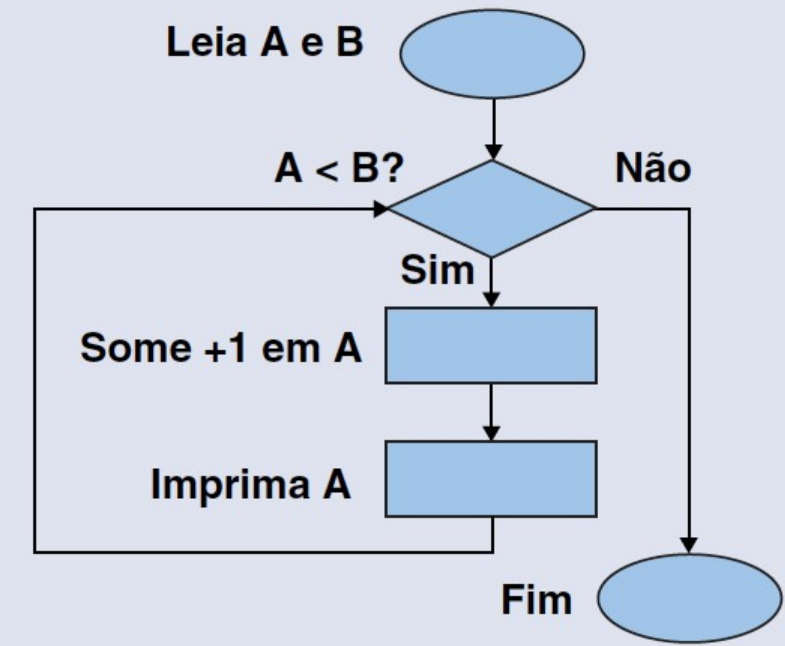
- Há casos em que é preciso que um bloco de comandos seja executado mais de uma vez se determinada condição for verdadeira:

```
enquanto condição faça  
    sequência de comandos;  
fim enquanto
```

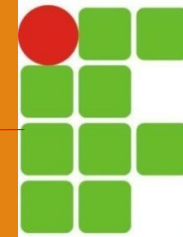
- Precisamos então de uma estrutura de repetição que permita executar um conjunto de comandos quantas vezes forem necessárias.
- De acordo com a condição, os comandos serão repetidos zero (se falsa) ou mais vezes (enquanto a condição for verdadeira). Essa estrutura normalmente é denominada laço ou loop.

Exemplo: pseudocódigo e fluxograma

```
Leia A e B;  
Enquanto A < B  
    A recebe A + 1;  
    Imprima A;  
Fim Enquanto
```



COMANDO DE REPETIÇÃO EM C



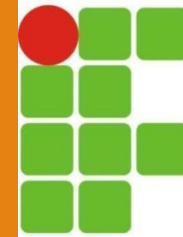
INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA

- Note que a sequência de comandos a ser repetida está subordinada a uma condição.
- Por condição entende-se qualquer expressão relacional (ou seja, que use os operadores >, <, >=, <=, == ou !=) que resulte em uma resposta do tipo verdadeiro ou falso.
- A condição pode ainda ser uma expressão que utiliza operadores:
 - Matemáticos : +, -, *, /, %
 - Relacionais: >, <, >=, <=, ==, !=
 - Lógicos: &&, ||
- Um laço infinito (ou loop infinito) é uma sequência de comandos em um programa de computador que sempre se repete, ou seja, infinitamente. Isso geralmente ocorre por algum erro de programação, quando:
 - Não definimos uma condição de parada.
 - A condição de parada existe, mas nunca é atingida.

```
X recebe 4;  
enquanto (X < 5) faça  
    X recebe X - 1;  
    Imprima X;  
fim enquanto
```

```
X recebe 4;  
enquanto (X < 5) faça  
    Imprima X;  
fim enquanto
```

COMANDO WHILE



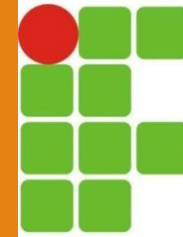
INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA

- Equivale ao comando “enquanto”
- A forma geral de um comando while é:

```
while (condição){  
    sequência de comandos;  
}
```

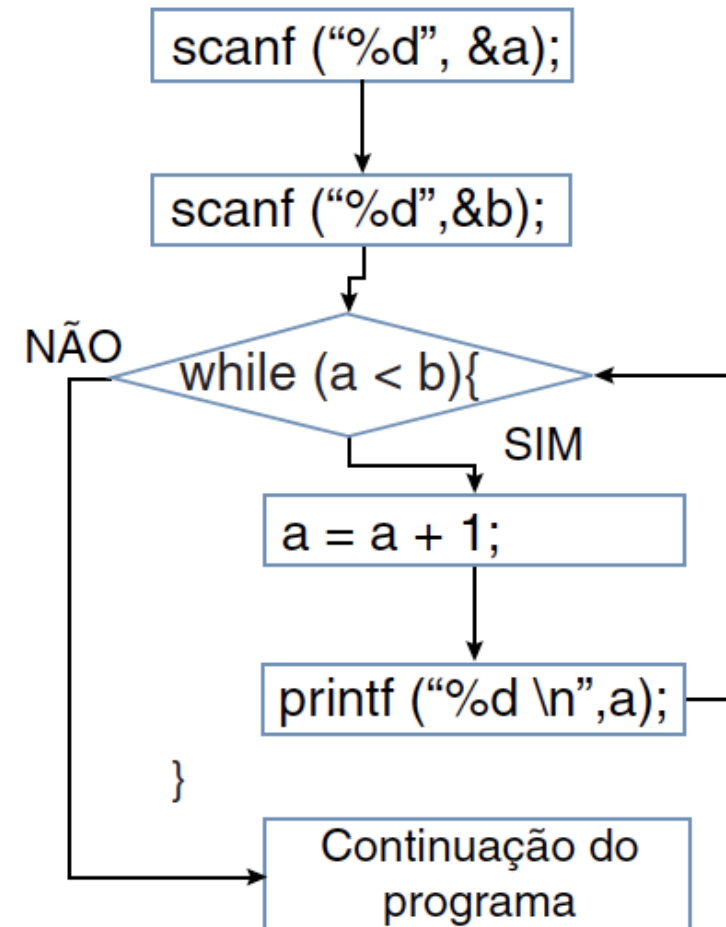
```
#include <stdio.h>  
#include <stdlib.h>  
int main(){  
    int a,b;  
    printf("Digite o valor de a: ");  
    scanf("%d",&a);  
    printf("Digite o valor de b: ");  
    scanf("%d",&b);  
    while (a < b){  
        a = a + 1;  
        printf("%d \n",a);  
    }  
    system("pause");  
    return 0;  
}
```

COMANDO WHILE



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int a,b;
    printf("Digite o valor de a: ");
    scanf("%d",&a);
    printf("Digite o valor de b: ");
    scanf("%d",&b);
    while (a < b){
        a = a + 1;
        printf("%d \n",a);
    }
    system("pause");
    return 0;
}
```



COMANDO FOR



- O comando for é muito similar ao comando while, visto anteriormente.
- Basicamente, o comando for é usado para repetir um comando ou uma sequência de comandos diversas vezes.
- A forma geral de um comando for é:

```
for (inicialização; condição; incremento) {  
    sequência de comandos;  
}
```

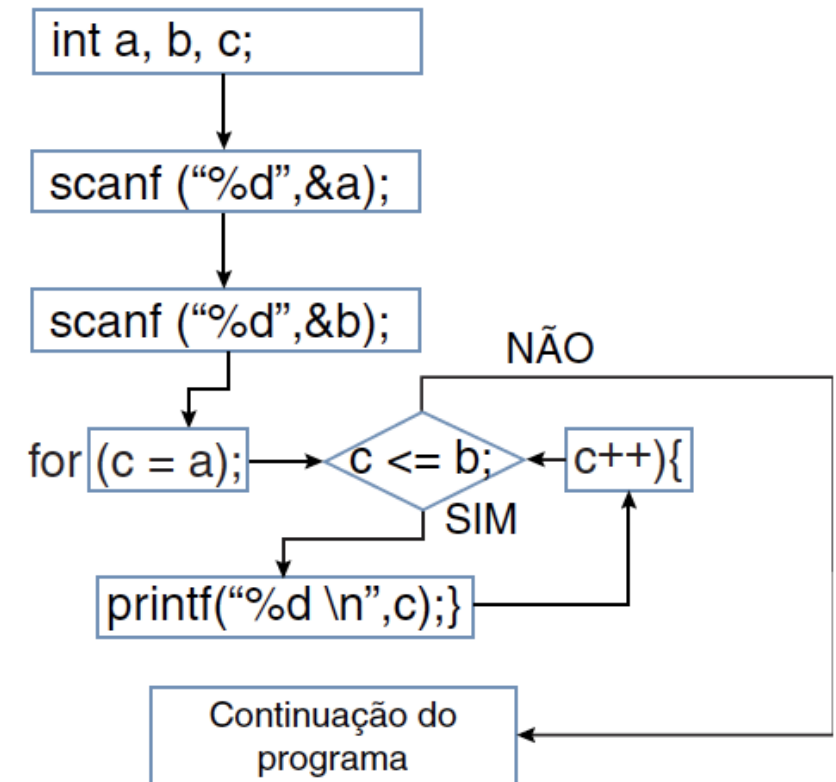
- Na execução do comando for, é realizada esta sequência de passos:
 - A cláusula inicialização é executada: nela as variáveis recebem um valor inicial para usar dentro do for.
 - A condição é testada:
 - Se a condição for considerada verdadeira (ou possuir valor diferente de zero), a sequência de comandos será executada. Ao final da sequência de comandos, o fluxo do programa é desviado para o incremento.
 - Se a condição for considerada falsa (ou possuir valor igual a zero), a sequência de comandos não será executada (fim do comando for).
 - Incremento: terminada a execução da sequência de comandos, ocorre a etapa de incremento das variáveis usadas no for. Ao final dessa etapa, o fluxo do programa é novamente desviado para a condição.

COMANDO WHILE

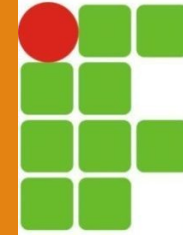


INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int a,b,c;
    printf("Digite o valor de a: ");
    scanf("%d",&a);
    printf("Digite o valor de b: ");
    scanf("%d",&b);
    for (c = a; c <= b; c++){
        printf("%d \n",c);
    }
    system("pause");
    return 0;
}
```



COMANDO WHILE



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA

Exemplo: for *versus* while

for

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i, s = 0;
    for(i = 1; i <= 10; i++){
        s = s + i;
    }
    printf("Soma = %d \n",s);
    system("pause");
    return 0;
}
```

while

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i, s = 0;
    i = 1
    while (i <= 10){
        s = s + i;
        i++;
    }
    printf("Soma = %d \n",s);
    system("pause");
    return 0;
}
```