

# Aula 5

## Estrutura de Repetição e Vetores – Exemplos de soluções

**Seiji Isotani, Rafaela V. Rocha**

[sisotani@icmc.usp.br](mailto:sisotani@icmc.usp.br)

[rafaela.vilela@gmail.com](mailto:rafaela.vilela@gmail.com)

**PAE: Armando M. Toda, Geiser Chalco**

[armando.toda@gmail.com](mailto:armando.toda@gmail.com)

[geiser.gcc@gmail.com](mailto:geiser.gcc@gmail.com)

# Estrutura de Repetição: PARA

// Pseudocódigo

**PARA**  $i \leftarrow \text{valorInicial}$  **ATÉ**  $\text{valorFinal}$  **PASSO**  $x$  **FAÇA**

Instrução\_1

Instrução\_2

....

Instrução\_n

**FIMPARA**

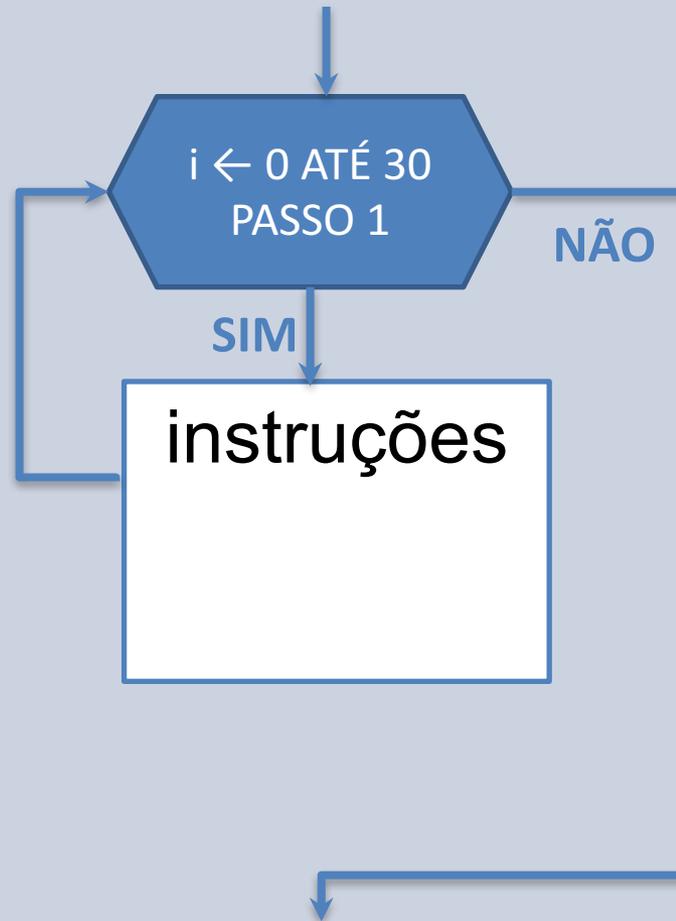
//Linguagem C:

```
for ( $i = \text{valor\_inicial}; \text{valor\_final};$   
 $\text{valor\_incremento}$ ) {  
    //Instrução_1  
    //Instrução_2  
    //....  
    //Instrução_n  
}
```

//exemplo:

```
for ( $i=1; i \leq 10; i++$ ){  
    //Instrução_n  
}
```

# Fluxograma: PARA



## Exercícios

1. Faça um algoritmo que exiba na tela uma contagem de 0 até 30, exibindo apenas os múltiplos de 3.
2. Faça um algoritmo que leia um conjunto de 10 valores, armazenando em um vetor. Uma vez lidos os valores, exibir na ordem inversa em que foram lidos os dados, ou seja, o último dado a ser exibido na tela deve ser o primeiro que foi lido.

# 1. Faça um algoritmo que exiba na tela uma contagem de 0 até 30, exibindo apenas os múltiplos de 3.

## Pseudocódigo

DECLARA i inteiro  
PARA i ← 1 ATÉ 30 PASSO 3 FAÇA  
    ESCREVA i  
FIMPARA

OU

DECLARA i inteiro  
PARA i ← 0 ATÉ 30 PASSO 1 FAÇA  
    SE (i%3 == 0)  
        ESCREVA i  
    FIMSE  
FIMPARA

## Linguagem C

i=i+3 ou i+=3

```
1 - int main() {  
2 -     for(int i=0; i<=30; i=i+3){  
3 -         printf("%d, ", i);  
4 -     }  
5 -     return 0;  
6 - }
```

```
1 - int main() {  
2 -     for(int i=0; i<=30; i++){  
3 -         if (i%3 == 0)  
4 -             printf("%d, ", i);  
5 -     }  
6 -     return 0;  
7 - }
```

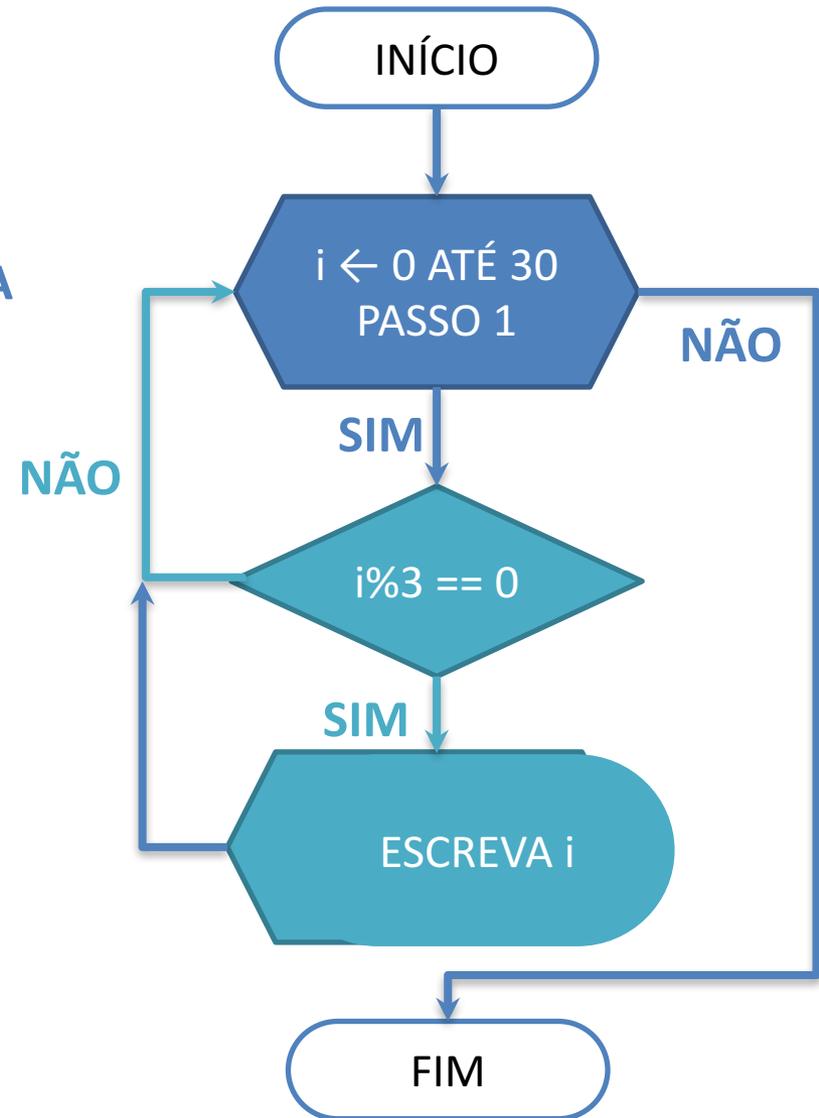
input Output

Success time: 0 memory: 10320 signal:0

0, 3, 6, 9, 12, 15, 18, 21, 24, 27, 30,

# Fluxograma

PARA  $i \leftarrow 0$  ATÉ 30 PASSO 1 FAÇA  
SE ( $i \% 3 == 0$ )  
ESCREVA  $i$   
FIMSE  
FIMPORA



2. Faça um algoritmo que leia um conjunto de 10 valores, armazenando em um vetor. Uma vez lidos os valores, exibir na ordem inversa em que foram lidos os dados, ou seja, o último dado a ser exibido na tela deve ser o primeiro que foi lido.

```

1  /|vetor[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
2  int main() {
3      int vetor[10];
4      for(int i=0; i<10; i++){
5          scanf("%d", &vetor[i]);
6      }
7      for(int i=9; i>=0; i--){
8          printf("%d, ", vetor[i]);
9      }
10     return 0;
11 }

```

OU

```

1  int main() {
2      int vetor[10];
3      for(int i=0; i<10; i++){
4          scanf("%d", &vetor[i]);
5      }
6      for(int i=0; i<10; i++){
7          printf("%d, ", vetor[9-i]);
8      }
9      return 0;
10 }

```

### Pseudocódigo

DECLARE v[10] inteiros

PARA  $i \leftarrow 0$  ATÉ 9 PASSO 1 FAÇA  
**LEIA v[i]**

FIMPARA

PARA  $i \leftarrow 9$  ATÉ 0 PASSO -1 FAÇA  
**IMPRIMA v[i]**

FIMPARA

input Output

Success time: 0 memory: 10320 signal:0  
10, 9, 8, 7, 6, 5, 4, 3, 2, 1,

## Erros comuns

**Leia  $x[10]$**  => só irá ler o valor da posição 9 do vetor

**Imprima  $(x[10], x[9], x[8]...)$**  => o ideal é usar uma estrutura de repetição. E se fossem 1000 números?

**$v[10] = v[1]$**

...

**$v[1] = v[10]$**  => neste caso  $v[1] = v[1]$ , pois a 1ª instrução foi  $v[10] = v[1]$

```

1 int main() {
2     for(int i=0; i<=30; i+3){
3         printf("%d, ", i);
4     }
5 }
```

correto:  **$i=i+3$**

**Pré-Incremento (++i): incrementa antes**  
**Pós-Incremento (i++): incrementa depois**

```
1 ▾ int main() {  
2     int a = 0 , b = 0;  
3     printf("%d, %d", ++a, b++);  
4 }
```

1      0

**O mesmo para:**

**Pré-Decremento (--i): decrementa antes**  
**Pós-Decremento (i--): decrementa depois**

```
1 ▾ int main() {  
2     int a = 0 , b = 0;  
3     printf("%d, %d", --a, b--);  
4 }
```

-1      0