

Disciplina: TCC-03.063 Prog. de Computadores III **Turma:** E-1 **Data:** ____/____/____
Professor: Leandro Augusto Frata Fernandes

Exercícios de Fixação Aulas 05 e 06

1. Construa um algoritmo (pseudocódigo e fluxograma) que determine e mostre o maior número entre vários fornecidos pelo usuário. A condição de parada é a entrada de um valor 0 (zero), ou seja, o algoritmo deve ficar escolhendo o maior até que a entrada seja igual a 0 (zero). Utilize a estrutura ENQUANTO ... REPETIR ... no algoritmo. Assuma que valores negativos nunca serão informados pelo usuário.

variáveis

inteiro: NumAtual, MaiorNum

início

```

1      MaiorNum ← 0
2      NumAtual ← -1

3      enquanto NumAtual <> 0 repetir
4          Ler NumAtual
5          se NumAtual > MaiorNum então
6              MaiorNum ← NumAtual
7          fim se
8      fim enquanto

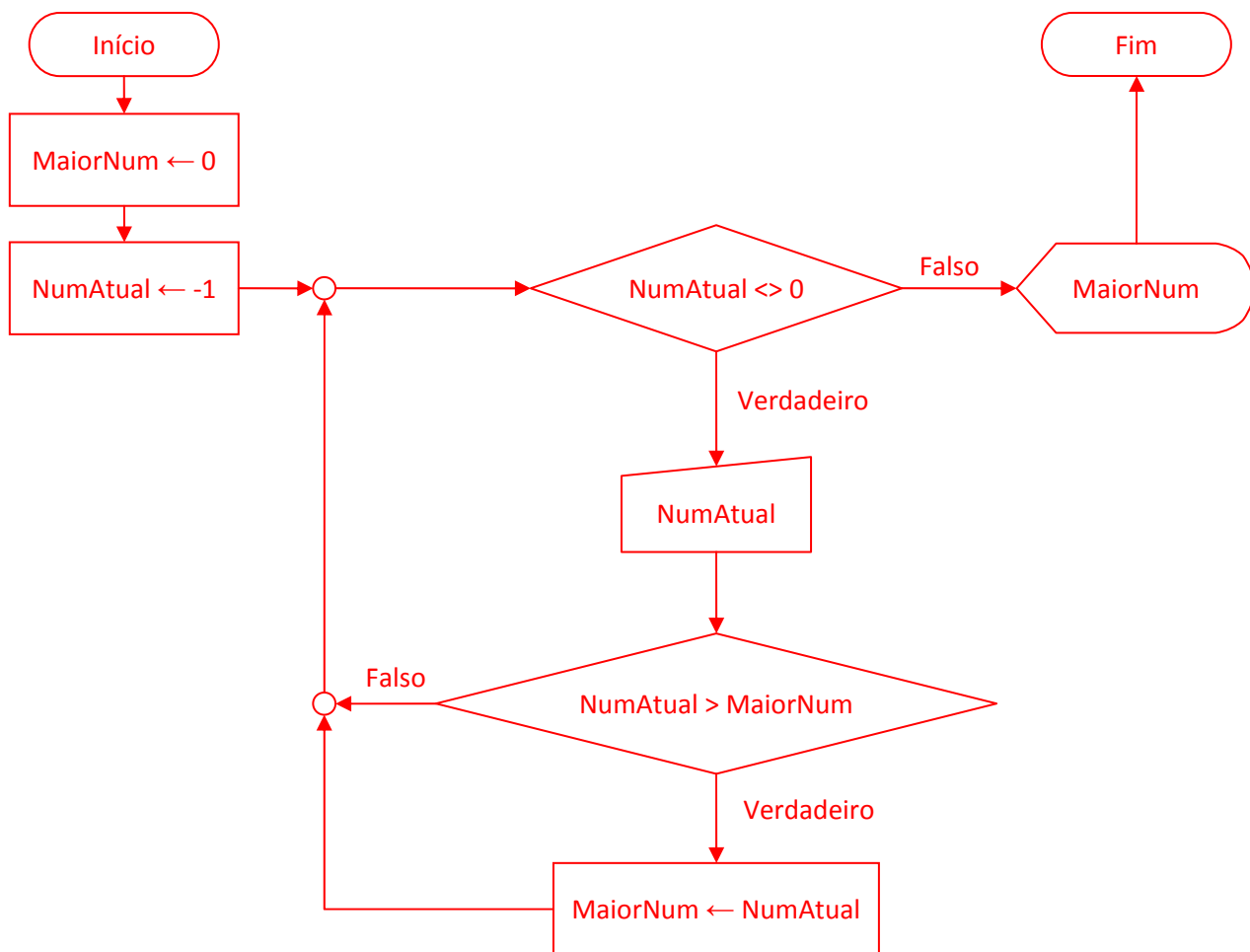
9      Mostrar MaiorNum
fim

```

A variável MaiorNum precisa ser inicializada com um valor menor que qualquer valor que possa ser informado pelo usuário. Neste caso, foi inicializada com 0 (zero) na Linha 1 do pseudocódigo.

A variável NumAtual precisa ser inicializada com qualquer valor diferente de 0 (zero), para que haja a possibilidade de entrar do laço ENQUANTO... REPETIR... Neste caso foi escolhido -1 (veja a Linha 2 do pseudocódigo). Note que o primeiro valor atribuído a NumAtual não afeta a escolha do maior número dentre os informados pelo usuário, pois o valor de inicialização é sobrescrito por um valor informado pelo usuário na Linha 4 do pseudocódigo, antes de ser feita a comparação com o maior número armazenado até o momento (Linha 5).

Outras soluções (algoritmos) equivalentes podem ser construídas. Teste sua solução fazendo o teste de mesa.



2. Construa um algoritmo (pseudocódigo) que leia 500 valores inteiros e positivos e:

- Encontre e mostre o maior valor.
- Encontre e mostre o menor valor.
- Calcule e mostre a média dos números lidos.

Utilize a estrutura ENQUANTO ... REPETIR ... no algoritmo.

variáveis

inteiro: Cont, NumAtual, MaiorNum, MenorNum, Soma, Media

início

```

1      MaiorNum ← -∞
2      MenorNum ← +∞
3      Soma ← 0

4      Cont ← 0
5      enquanto Cont < 500 repetir
6          Ler NumAtual

7          se NumAtual > MaiorNum então
8              MaiorNum ← NumAtual
9          fim se

10         se NumAtual < MenorNum então
11             MenorNum ← NumAtual
12         fim se

13         Soma ← Soma + NumAtual
14         Cont ← Cont + 1
15     fim enquanto

16     Media ← Soma / Cont

17     Mostrar MaiorNum
18     Mostrar MenorNum
19     Mostrar Media
fim

```

Assim como no exercício anterior, a variável MaiorNum precisa ser inicializada com um valor menor que qualquer valor que possa ser informado pelo usuário, para que este valor inicial possa ser “vencido” por qualquer outro número informado pelo usuário. A variável MenorNum, por sua vez, precisa receber um valor maior que qualquer valor que possa ser informado pelo usuário, para que o valor inicial também possa ser “vencido”.

Na solução apresentada optei por usar $-\infty$ e $+\infty$ para inicializar MaiorNum e MenorNum, respectivamente (veja as Linhas 1 e 2 do pseudocódigo).

Outras soluções (algoritmos) equivalentes podem ser construídas. Teste sua solução fazendo o teste de mesa.

3. Reescreva o algoritmo desenvolvido no Exercício 1 utilizando a estrutura de repetição REPETIR ... ENQUANTO ... no lugar de ENQUANTO ... REPETIR ...

variáveis

inteiro: NumAtual, MaiorNum

início

1 MaiorNum \leftarrow 0

2 **repetir**

3 **Ler** NumAtual

4 **se** NumAtual > MaiorNum **então**

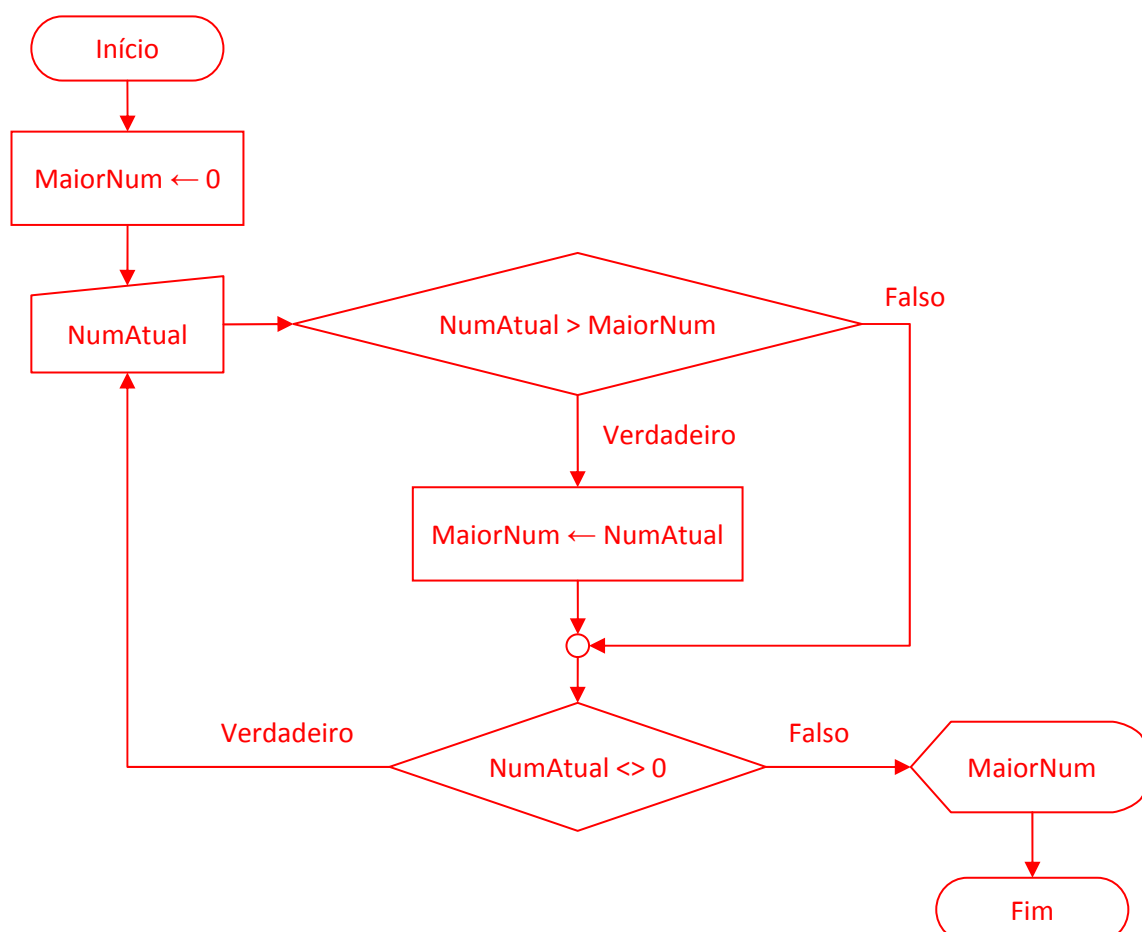
5 MaiorNum \leftarrow NumAtual

6 **fim se**

7 **enquanto** NumAtual \neq 0

8 **Mostrar** MaiorNum

fim



4. Utilizando a estrutura PARA... ATÉ... REPETIR..., construa um algoritmo (pseudocódigo) que exiba os números ímpares entre 100 e 200.

Dica: a função $\text{mod}(X,Y)$ retorna o resto da divisão de X por Y.

Solução utilizando passo diferente de 1 (um)

variáveis

inteiro: NumAtual

início

```
1   para NumAtual ← 101 até 199 passo 2 repetir
2       Mostrar NumAtual
3   fim para
fim
```

Solução utilizando passo igual a 1 (um)

variáveis

inteiro: NumAtual

início

```
1   para NumAtual ← 100 até 200 repetir
2       se mod( NumAtual, 2 ) <> 0 então
3           Mostrar NumAtual
4       fim se
5   fim para
fim
```

5. Construa um algoritmo (pseudocódigo) que calcule a média de N números reais informados pelo usuário, onde o valor de N também é informado no início da execução. Utilize a estrutura PARA... ATÉ... REPETIR... Antes do término da execução do algoritmo, mostre a média calculada.

variáveis

inteiro: i, N

real: NumAtual, Soma, Media

início

```

1      repetir
2          Ler N
3      enquanto N <= 0

4          Soma ← 0
5      para i ← 1 até N repetir
6          Ler NumAtual
7          Soma ← Soma + NumAtual
8      fim para

9      se N > 0 então
10         Media ← Soma / N
11     se não
12         Media ← 0
13     fim se

14     Mostrar Media
fim

```

Neste exercício é importante notar que não sabemos *a priori* a quantidade de valores que serão informados pelo usuário. Saberemos esta quantidade (N) apenas em tempo de execução do algoritmo. Por conta deste detalhe, as Linhas 1 a 3 do pseudocódigo garantem que a quantidade de números informados pelo usuário será sempre maior ou igual a 0 (zero). Afinal, informar uma quantidade negativa de número não faz sentido.

Também, precisamos ser mais cuidadosos ao calcular a média. Precisamos testar se a quantidade de valores é diferente de 0 (zero), a fim de evitar divisões por zero. Isso é feito na Linha 9 do pseudocódigo.