

Alguns exemplos resolvidos: Lista 03

**12. Efetuar a leitura de cinco números inteiros e identificar o maior e o menor. Não execute a ordenação de valores.**

Base de Conhecimento (Até o Conteúdo 03)

```
//Diretivas de Pré-processamento (Obrigatórias)
#include <stdio.h>
#include <stdlib.h>
//Obrigatório. Função principal: indica o início da execução do programa
int main ()
{
    //Declaração de Variáveis Locais.
    int ...;
    float ....;
    //Obrigatório. Comandos para resolução do problema
    //Possíveis mensagens para indicar as entradas
    printf("...");
    //Possíveis comandos para coletar valores
    scanf("...", &...);

    // Quando necessário, utilizar os comandos para o controle de fluxo
    if (num>5)
        printf("O número digitado é maior que 5");
    else
        printf("O número digitado é menor que 5");

    //Possíveis mensagens para indicar a(s) saída(s)
    printf("....", ...);
    //Opcional. Comando para interromper momentaneamente o programa
    printf("\n");
    system("PAUSE");
    //Retorno ao SO o status do programa
    return 0;
} //Indica o final do programa.
```

---

**Rascunho para resolver o exercício**

Algoritmo Ex12\_lista03.c

**início**

**//Variáveis para permitir as Entrada(s):**

num\_1, num\_2, num\_3, num\_4, num\_5

**//Processamento(s): (Identificar o maior número)**

se (num\_1>num\_2 && num\_1>num\_3 && num\_1>num\_4 && num\_1>num\_5)  
    escreva ("numero maior é:", num\_1);

senão se (num\_2>num\_3 && num\_2>num\_4 && num\_2>num\_5)  
    escreva ("numero maior é:", num\_2);

```
senão se (num_3>num_4 && num_3>num_5)
    escreva (“numero maior é:”, num_3);
```

```
senão se (num_4>num_5)
    escreva (“numero maior é:”, num_4);
```

```
senão
    escreva (“numero maior é:”, num_5);
```

### **//Processamento(s): (Identificar o menor número)**

```
se (num_1<num_2 && num_1<num_3 && num_1<num_4 && num_1<num_5)
    escreva (“numero menor é:”, num_1);
```

```
senão se (num_2<num_3 && num_2<num_4 && num_2<num_5)
    escreva (“numero menor é:”, num_2);
```

```
senão se (num_3<num_4 && num_3<num_5)
    escreva (“numero menor é:”, num_3);
```

```
senão se (num_4<num_5)
    escreva (“numero menor é:”, num_4);
```

```
senão
    escreva (“numero menor é:”, num_5);
```

### **//Saída(s):**

// escrita do maior número, a partir da condição atendida

// escrita do menor número, a partir da condição atendida

**fim**

---

## **Codificação do Rascunho: Linguagem C**

//Diretivas de Pré-processamento (Obrigatórias)

#include <stdio.h>

#include <stdlib.h>

//Obrigatório. Função principal: indica o início da execução do programa

int main ()

{

//Declaração de Variáveis Locais.

int num\_1, num\_2, num\_3, num\_4, num\_5;

//Obrigatório. Comandos para resolução do problema

//Possíveis mensagens para indicar as entradas

printf("\nDigite cinco números:");

printf("\nPrimeiro valor:");

scanf("%d",&num\_1);

printf("\nSegundo valor:");

scanf("%d",&num\_2);

printf("\nTerceiro valor:");

```

scanf("%d",&num_3);
printf("\nQuarto valor:");
scanf("%d",&num_4);
printf("\nQuinto valor:");
scanf("%d",&num_5);
//Identificação do maior número e a saída
if (num_1>num_2 && num_1>num_3 && num_1>num_4 && num_1>num_5)
    printf("\n numero maior é: %d", num_1);
else if (num_2>num_3 && num_2>num_4 && num_2>num_5)
    printf("\n numero maior é: %d", num_2);
else if (num_3>num_4 && num_3>num_5)
    printf("\n numero maior é: %d", num_3);
else if (num_4>num_5)
    printf("\n numero maior é: %d", num_4);
else
    printf("\n numero maior é: %d", num_5);

//Identificação do menor número e a saída
if (num_1<num_2 && num_1<num_3 && num_1<num_4 && num_1<num_5)
    printf("\n numero menor é: %d", num_1);
else if (num_2<num_3 && num_2<num_4 && num_2<num_5)
    printf("\n numero menor é: %d", num_2);
else if (num_3<num_4 && num_3<num_5)
    printf("\n numero menor é: %d", num_3);
else if (num_4<num_5)
    printf("\n numero menor é: %d", num_4);
else
    printf("\n numero menor é: %d", num_5);

//system("PAUSE");
//Retorno ao SO o status do programa
return 0;
} //Indica o final do programa.

```

---

**7. Construa um algoritmo para apresentar procedência de um produto. Para isso, o algoritmo deve ler um código de origem e comparar com os valores definidos na tabela a seguir. Caso o código informado não esteja listado, o produto deve ser definido como importado.**

Código:de origem	Procedência
1	Sul
2	Norte
3	Leste
4	Oeste
5 ou 6	Nordeste
7, 8 ou 9	Sudeste
10 ou 11	Centro-Oeste
12, 13, 14 ou 15	Nordeste
Qualquer outro código	Importado

```

//Diretivas de Pré-processamento (Obrigatórias)
#include <stdio.h>
#include <stdlib.h>
//Obrigatório. Função principal: indica o início da execução do programa
int main ()
{
    int prod;
    printf ("\nDigite o código do produto: \n");
    scanf ("%d",&prod);
    switch(prod)
    {
        case 1: printf("\nSul"); break;
        case 2: printf("\nNorte"); break;
        case 3: printf("\nLeste"); break;
        case 4: printf("\nOeste"); break;
        case 5:
        case 6:
        case 12 ... 15: printf("\nNordeste"); break;
        case 7 ... 9: printf("\nSudeste"); break;
        case 10:
        case 11: printf("\nCentro-Oeste"); break;
        default: printf("Importado"); break;

    }

    //Opcional. Comando para interromper momentaneamente o programa
    //system("PAUSE");
    //Retorno ao SO o status do programa
    return 0;
}
//Indica o final do programa.

```