

C202-A

Algoritmos e Estruturas de Dados I

08 – Variáveis Indexadas (unidimensionais - vetores)

Prof. Edson J. C. Gimenez
soned@inatel.br

2019/Sem1

Escreva um programa para ler 100 números e mostrar a média aritmética desses 100 números.

Dentre os elementos fornecidos, mostrar quantos estão acima da média, quantos estão abaixo da média, e quantos são iguais a média.

Para a solução deste problema, é necessário que se tenha uma estrutura de dados que contenha todos os 100 números armazenados, de tal forma que possam ser referenciados pelo conjunto de números (NUM), acessando cada número individualmente.

NOTAS

Valor	5.0	3.0	8.5	7.6	9.0	5.5
Posição	0	1	2	3	98	99

Este tipo de estrutura de dados é conhecida como **VARIÁVEL INDEXADA**.

VARIÁVEIS INDEXADAS

As variáveis indexadas (ou compostas) referenciam grupos de variáveis do mesmo tipo (homogêneas), pelo mesmo nome.

São individualizadas entre si através de uma posição dentro desse conjunto.

Ao número de índices necessários à localização de um elemento dentro de uma variável indexada dá-se o nome de dimensão.

Vetores:

Variáveis indexadas homogêneas unidimensionais (único índice).

Matrizes:

Variáveis indexadas homogêneas multidimensionais (dois ou mais índices).

VARIÁVEIS INDEXADAS HOMOGÊNEAS UNIDIMENSIONAIS (VETORES)

Conjunto de dados homogêneos referenciados por um mesmo nome que necessitam de um único índice para ter seus elementos individualizados.

O vetor possui uma única dimensão, sendo a primeira posição a posição 0.

SINTAXE DA VARIÁVEL INDEXADA HOMOGÊNEA UNIDIMENSIONAL (VETOR)

tipo básico lista de variáveis [**tamanho**];

sendo:

- **tipo básico** representa qualquer um dos tipos básicos (int, float, double, char, etc.);
- **tamanho** representa a quantidade de elementos que vai compor o vetor;
- **lista de variáveis** representa o(s) nome(s) de variáveis indexadas que terão o tamanho e o tipo definidos;

Exemplo 1: declaração de vetores

```
int y[5], x[100];  
char nome[30];  
float notas[60];
```

y → vetor com 5 posições de inteiros.

x → vetor com 100 posições de inteiros.

nome → vetor com 30 posições de “char” (caracteres).

Notas → vetor com 60 posições de “float” (ponto flutuante).

Exemplo 2:

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    float soma=0, x[10];
    for (i = 0; i < 5; i++)
    {
        cout<<"Digite o elemento x("<<i<<")= ";
        cin>>x[i];
        soma = soma + x[i];
    }
    cout<<endl<<"A soma dos elementos = "<<soma<<endl;
    cout<<"A média dos elementos = "<<soma/5<<endl;
    return 0;
}
```

C:\Users\edsonjcg\Documents\2019_Sem1

```
Digite o elemento x(0)= 4
Digite o elemento x(1)= 1
Digite o elemento x(2)= 3
Digite o elemento x(3)= 2
Digite o elemento x(4)= 5
```

```
A soma dos elementos = 15
A media dos elementos = 3
```

Exemplo 3:

```
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
    int i, id[10], ma, me;
    string no[10];
    //entrada
    for (i = 0; i < 5; i++){
        cout<<"Candidato "<< i+1<<endl;;
        cout<<"Nome: ";
        cin>>no[i];
        do{
            cout<<"Idade: (>=17 e <60 anos) ";
            cin>>id[i];
        }while(id[i]<17 || id[i]>=60);
        if(i==0)
            ma = id[0]; me = id[0];
        else {
            if(id[i]>ma) ma=id[i];
            if(id[i]<me) me=id[i];
        }
    } //fim do for
```

→ CONTINUA....

C:\Users\edsonjcg\Documents\2019_Sem1

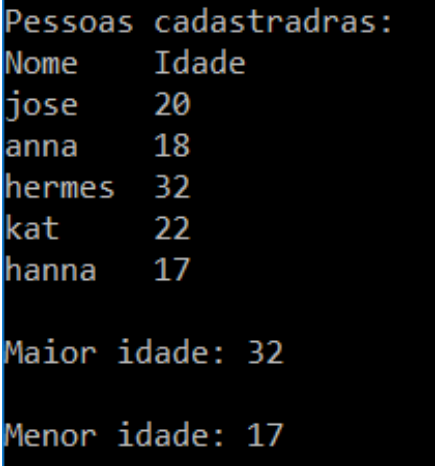
```
Candidato 1
Nome: jose
Idade: (>=17 e <60 anos) 20
Candidato 2
Nome: anna
Idade: (>=17 e <60 anos) 18
Candidato 3
Nome: hermes
Idade: (>=17 e <60 anos) 32
Candidato 4
Nome: kat
Idade: (>=17 e <60 anos) 22
Candidato 5
Nome: hanna
Idade: (>=17 e <60 anos) 17
```

Exemplo 3:

```

:
:
//saida
system("cls");
cout<<"Pessoas cadastradas: "<<endl;
cout<<"Nome"<<"\t"<<"Idade"<<endl;
for (i = 0; i < 5; i++)
{
    cout<<no[i]<<"\t"<<id[i]<<endl;
}
cout<<endl<<"Maior idade: "<<ma<<endl;
cout<<endl<<"Menor idade: "<<me<<endl;
return 0;
}

```



```

C:\Users\edsonjcg\Documents\20
Pessoas cadastradas:
Nome      Idade
jose      20
anna      18
hermes    32
kat       22
hanna     17

Maior idade: 32

Menor idade: 17

```

Exercício 5.1) O que será impresso pelo trecho de código a seguir?

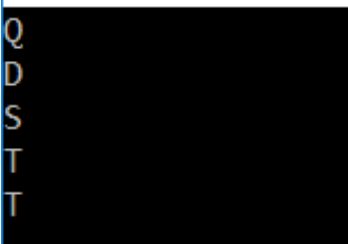
```

#include <iostream>
using namespace std;
int main()
{
    int i;
    int ve [6];
    char ca [6];

    ve[0] = 1;  ve[1] = 1;  ve[2] = 2;
    ve[3] = 5;  ve[4] = 0;  ve[5] = 1;
    ca[0] = 'S'; ca[1] = 'T'; ca[2] = 'Q';
    ca[4] = 'S'; ca[5] = 'D';
    for (i = 2; i <= 4; i = i + 1)
        cout << ca[ve[i]] << endl;
    cout << ca[ve[i]] << endl;
    cout << ca[ve[ve[3]]] << endl;

    return 0;
}

```



```

C:\Users\edsonjcg\Docu
Q
D
S
T
T

```

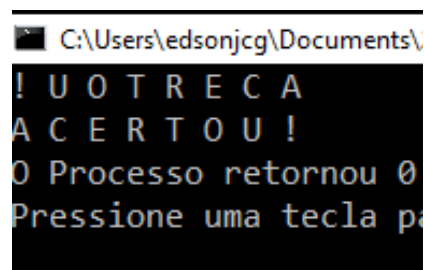
Exercício 5.2) Dado o vetor **CRR** literal (de caracteres) abaixo:

CRR →	!	U	O	T	R	E	C	A
	0	1	2	3	4	5	6	7

Qual será a sua configuração depois de executados os comandos:

```

:
for (i = 1; i <= 3; i = i + 1)
{
    aux = crr[i];
    crr[i] = crr[6-i+1];
    crr[6-i+1] = aux;
}
aux = crr[0];
crr[0] = crr[7];
crr[7] = aux;
:
    
```



Exercício 6.3) Se você tem um vetor com 100 valores numéricos:

- Como seria a declaração de um vetor para armazenar esses 100 números?
- Qual seria o bloco de comandos para ler os 100 números e armazená-los neste vetor?
- Qual seria o bloco de comandos para calcular a média dos 100 números deste vetor?
- Qual seria o bloco de comandos para calcular o maior número dentre os 100 valores deste vetor?
- Qual seria o bloco de comandos para imprimir este vetor de 100 posições?

```
#include <iostream>
using namespace std;
int main(){
    float vet[10], media, maior;
    int i;

    // leitura
    for (i=0; i<10; i++){
        cout<<"Posicao: "<<i<<" ";
        cin>> vet[i];
    }

    // Calculo da media
    for (i=0; i<10; i++)
        media = media + vet[i];
    cout<<"\nMedia dos elementos: "<< media/10;

    //Maior elemento
    maior = vet[0];
    for (i=1; i<10; i++)
        if (vet[i]>maior)
            maior = vet[i];
    cout<<"\n\nMaior elemento: "<< maior<<endl;

    // Mostrar todos
    for (i=0; i<10; i++)
        cout<<"\nPosicao: "<<i<<" ": "<<vet[i];

    return 0; } // fim main
```

```
C:\Users\edsonjcg\Documents\Projeto
Posicao: 0: 9
Posicao: 1: 8
Posicao: 2: 7
Posicao: 3: 6
Posicao: 4: 5
Posicao: 5: 4
Posicao: 6: 3
Posicao: 7: 2
Posicao: 8: 1
Posicao: 9: 0

Media dos elementos: 4.5

Maior elemento: 9

Posicao: 0: 9
Posicao: 1: 8
Posicao: 2: 7
Posicao: 3: 6
Posicao: 4: 5
Posicao: 5: 4
Posicao: 6: 3
Posicao: 7: 2
Posicao: 8: 1
Posicao: 9: 0

-----
Process exited after 10.44 s
Pressione qualquer tecla para
```

Exercício 3 Elabore um programa que leia uma sequência de **n** números ($n \leq 100$), depois imprima-os em ordem inversa (do que foi digitado).

```
1  #include <iostream>
2  #include <locale>
3  using namespace std;
4  int main ( )
5  { //declaração das variáveis
6      double vet[100]; //vetor que armazenará os 100 números
7      int i, N;
8      setlocale(LC_ALL, "portuguese");
9      cout<<"Quantos números? ";
10     cin>>N;
11     while(N<1 || N>100)
12     {
13         cout<<"Qde deve ser >= 1 e <= 100 - digite novamente: ";
14         cin>>N;
15     }
16     // repetição para ler os 100 números
17     for( i=0; i<=N-1; i++) // de 0 a N-1 = N repetições
18     {
19         cout<<"Digite o "<< i+1 <<"º valor: ";
20         cin>> vet[i];
21     }
22     //repetição para mostrar os N números, em ordem inversa
23     cout<<endl<<"Valores em ordem inversa:";
24     for( i= N-1; i>=0; i--)
25         cout<<endl<<vet[i];
26     return 0;
27 }
```


Exercício 4) Escreva um programa que leia um conjunto A de 10 elementos numéricos, calcule e mostre o valor de S, sendo $S = (A_0 - A_9)^2 + (A_1 - A_8)^2 + (A_2 - A_7)^2 + (A_3 - A_6)^2 + (A_4 - A_5)^2$.

```

1  #include <iostream>
2  #include <locale>
3  #include <cmath>
4  using namespace std;
5  int main ( )
6  { //declaração das variáveis
7      double A[10], S, termo;
8      int i;
9      setlocale(LC_ALL, "portuguese");
10     // repetição para ler os 10 números
11     for( i=0; i<=9; i++) // de 0 a 9 = 10 repetições
12     {
13         cout<<"Digite o "<< i+1 <<"º valor: ";
14         cin>> A[i];
15     }
16     S = 0; //inicializar S com 0
17     //repetição para calcular S
18     for( i=0; i<=4; i++)
19     {
20         termo = A[i] - A[9-i];
21         S = S + pow( termo, 2 );
22     }
23     cout<<"\n S = "<<S<<endl;
24
25     return 0;
26 }

```

```

C:\Users\edsonjc...  [
Digite o 1º valor: 1
Digite o 2º valor: 2
Digite o 3º valor: 3
Digite o 4º valor: 4
Digite o 5º valor: 5
Digite o 6º valor: 1
Digite o 7º valor: 2
Digite o 8º valor: 3
Digite o 9º valor: 4
Digite o 10º valor: 5

S = 40

O Processo retornou 0
o de execução : 9.226 s
Pressione uma tecla pa
tinuar...

```

Exercício 5) Escreva um programa que leia e armazene um conjunto de N elementos numéricos em um vetor X, sendo $N \leq 50$, e depois mostre:

- quantos elementos são maiores ou iguais a zero e quantos são menores que zero;
- qual a posição (ou em quais posições), se encontra o valor 0 (zero);

```

1  #include <iostream>
2  #include <locale>
3  using namespace std;
4  int main ( )
5  { //declaração das variáveis
6      double X[50];
7      int i, N, cma, cme;
8      setlocale(LC_ALL, "portuguese");
9      cout<<"Quantos elementos? ( N>=1 e N<=50): ";
10     cin>> N;
11     while( N<1 || N>50)
12     {
13         cout<<"N deve estar entre 1 e 50 - digite novamente: ";
14         cin>> N;
15     }
16     // repetição para ler os N números
17     cma = 0; cme = 0;
18     for( i=0; i<N; i++)
19     {
20         cout<<"Digite o "<< i+1 <<"º valor: ";
21         cin>> X[i];
22         if( X[i]>=0) cma++;
23         else cme++;
24     }
25     cout<<"\nElementos >= 0: "<< cma <<" elementos.";
26     cout<<"\nElementos < 0: "<< cme <<" elementos.";
27     cout<<"\nPosição dos elementos iguais a zero no vetor:";
28     //repetição para testar se o elemento é 0
29     for( i=0; i<N; i++)
30         if( X[i] == 0) cout<<"\t"<<i;
31
32     return 0;
33 }

```