

**2 ED**

# **Leitura com getline() em vetores**

*Prof. Dr. Eliseu LS*

# **Conteúdo da Aula**

## **Revisão de vetores e programação**

- Prática de Vetores explícitos e não explícitos
- Criação de sub rotinas
- Leitura de strings com sub rotinas
- Estrutura de Menu de Controle

# ACHANDO O MAIOR E MENOR VALOR DE UM VETOR

Você pode também declarar vetores em forma de lista, neste caso os vetores serão alimentados no ato da declaração do vetor. Por exemplo: `int a [ ] = {1,2,3,4}`, `string b [ ] = {"a","b","c","d"}`, `char sex [ ] = {'F','M','F'}`, `bool x [ ] = {true, false, true, false}`.

Considere o vetor a seguir:

```
int xy [ ] = {1, 5, 10, 45, 3 , 9};
```

Vamos fazer um código para retornar o maior valor e outro para retornar o menor valor desta lista.

**Atenção:** 1) Indente/Organize o Código.  
2) Compile e execute no IDE.

## PROGRAMA 1

```
#include "iostream"
#include "math.h"
#include "cstdlib"
using namespace std;
int xy [ ] = { 1, 5, 10, 45, 3 , 9 };
void mostrar ( )
{ int maior, menor;
  int tot = sizeof(xy)/sizeof(int);
  for (int i = 0; i< tot; i++)
  {
    if ( xy[i] > maior || i == 0 ) maior = xy[i];
    if ( xy[i] <menor || i == 0 ) menor = xy[i];
  }
  cout << "\nMaior:" << maior;
  cout << "\nMenor" << menor << endl;
  system("pause");
}

int main () { mostrar ( ); return 0; }
```

# EXIBIR A MÉDIA DE UM VETOR

Você pode também realizar operações lógicas ou aritméticas com vetores, Considere o vetor a seguir:

```
int vet [ ] = {1,3,4,-1,0,4,19};
```

Vamos fazer um código para retornar a média dos valores desta lista.

**Atenção:** 1) Indente/Organize o Código.  
2) Compile e execute no IDE.

## PROGRAMA 2

```
#include "iostream"
#include "math.h"
#include "cstdlib"
using namespace std;
int vet [ ] = {1,3,4,-1,0,4,19};
float valorMedia() {
    float media = 0;
    int tot = sizeof(vet)/sizeof(int);
    for (int i = 0; i< tot; i++)
    { media += vet[i]; }

    media /= tot;
    return ( media ); }

int main () {
    float media = valorMedia();
    cout << endl << media << endl;
    system("pause"); }
```

# FAZENDO UMA TABELA COM VETORES

**PROGRAMA 3:** Construindo uma tabela a partir de 3 vetores, nome, idade e salário. O void novaLinha( ) faz a leitura e o void mostrar ( ) exibe os dados da tabela. **Insira as bibliotecas, indente, compile e execute o código.**

linha	Nome	Ida	Salário
0	Francilino	25	2500
1	Benedicto	40	3000
2	Ruanito	19	2500
3	Mariano	30	5000

```
int linha = -1;  
string nome [ 4 ]; int idade [4 ]; double salario [4];
```

```
void novaLinha(string nom, int ida, double sal){  
    linha ++;  
    string nom_; nom_ = nom;  
    nome[linha] = nom_;
```

```
    idade[linha] = ida;  
    salario[linha]=sal; }
```

```
void mostrar ( )    {  
    for ( int i = 0; i<=linha; i++)  
    { cout << endl << nome[i] << " - ";  
      cout << idade[i] << " - " << salario [i] ; }  
    system("pause"); }
```

# CRIANDO AS FUNÇÕES DE LEITURA / getline ( )

Agora altere o código da função main() novamente, faça funções de leitura para cada coluna da tabela e execute a partir da função main();

```
string lerNome() { string nome; cout << "\nNome:"; cin.ignore(); getline(cin, nome); return nome; }
```

```
int lerIdade() { int idade; cout << "\nIdade:"; cin >> idade; return idade; }
```

```
double lerSalario() { double sal; cout << "\nSalário:"; cin >> sal; return sal; }
```

# Lendo várias linhas dos vetores através de menu

```
int main()
{
    string nome_; int idade_; double salario_;

    int tecla;

    MENU:
        system("clear");
        cout << "* Menu *\n1 ler\n2 exibir\n3
sair\nitem:";
        cin >> tecla;
```

```
    switch ( tecla )
    {
        case 1: nome_ = lerNome();
                idade_ = lerIdade();
                salario_ = lerSalario();
                novaLinha ( nome_, idade_,
salario_);
                break;

        case 2: mostrar ();
                break;

        case 3: exit(0); break;
    }

    goto MENU;
    return 0; }
```

# Tarefa de Fixação / Avaliação

I- Faça o algoritmo para calcular o valor de uma prestação em atraso com base na multa de 2% e também do juros mensal de 1% sobre o valor principal. Declare vetores explícitos para entrada do valor da prestação, valorprincipal [ ] e da quantidade de dias em atraso: dias [ ]. Armazene os resultados no vetor ValorPagar [ 5 ]. Ao final imprima a tabela com os vetores valorPrincipal [], dias [] e valorpagar[] . Faça um menu de controle.

II - Crie um algoritmo que tenha um vetor explícito para armazenar 5 idades e um vetor não explícito para armazenar 5 nomes. Ao final Exiba o nome da pessoa de maior idade, menor idade, a média das idades e o saldo das idades.

NOTA: Entrega obrigatória individual juntamente com o enunciado e o print da tela de execução do programa em um único arquivo no formato DOCX ou PDF pelo MS-Teams.