

2

Usando Vetores em C/C++

Prof. Dr. Eliseu LS

ESTRUTURAS DE ARMAZENAMENTO

Estruturas **homogêneas** permitem o armazenamento de apenas um datatype por vez, como é o caso das matrizes, vetores, variáveis e constantes. Estruturas **heterogêneas** permitem o armazenamento de vários datatypes em uma única estrutura, como é o caso dos registros (structs), das tabelas em banco de dados, classes, etc.



Estruturas Homogêneas - Vetor (ARRAY)

VETOR é uma MATRIZ do tipo LISTA homogênea de dados, isto é, possui apenas uma dimensão apenas, onde cada elemento pode ser acessado pelo seu índice ou sua posição. Vetores podem ser declarados de forma explícita onde os dados já são inseridos na declaração ou de forma não explícita onde os dados serão lidos posteriormente:

[FORMA EXPLÍCITA]

define-se o nome e os dados,
o tamanho dependerá dos dados.

```
int x[ ] = { 1, 0, -1 };
```

n = total de elementos = 3

1 está na posição 0

0 está na posição 1

-1 está na posição (n - 1) que é 2

Vetor X []	
posição	elemento
0	1
1	0
2	-1

[NÃO EXPLÍCITA]

define-se o nome e o
tamanho somente. Os dados
são inseridos depois.

```
int x[3];  
x[0] = 5;  
x[1] = 10;  
x[2] = 3;
```

```
string nome[3];  
int i = 1;  
nome[i] = "Maria";  
i = 2;  
nome[i] = "João";
```

Vetores Explícitos (Listas) & Sizeof()

Os vetores explícitos são declarados sem tamanho, já que seu tamanho será definido pelo quantidade de elementos inseridos. Para descobrir o tamanho desse vetor, usa-se a função `sizeof ()`, você também pode fazer cálculos e outras operações com elementos do vetor.

PROGRAMA 1

Este exemplo cria o vetor `x []` com `n` elementos, em seguida descobre o número de elementos (`n`), `sizeof(x)` é o tamanho total do vetor `x`, `sizeof(int)` é o tamanho de um elemento do tipo `int`. cria uma variável `aux` para armazenar a soma do quadrado do elemento do vetor com ele mesmo. O loop `for` é utilizado para armazenar e exibir o valor de `aux`.

```
#include "iostream"
#include "math.h"
using namespace std;
int main () {
int x[ ] = { 1, 0 , -1 };
int n = sizeof(x)/sizeof(int); // total de
elementos
int aux = 0;
for (int i = 0; i < n ; i++)
{ aux = pow( x [ i ], 2 ) + x [ i ];
  cout << aux << endl;
}
return 0; }
```

Simulando o programa 1 - vetor explícito

n	i	x[i]	aux
3	0	$x[0] = 1$	$1^2 + 1 = 2$
3	1	$x[1] = 0$	$0^2 + 0 = 0$
3	2	$x[2] = -1$	$-1^2 + -1 = 0$

Os valores calculados de aux são respectivamente: 2 , 0, 0

Vetores não Explícitos (Necessário leitura)

Os vetores não explícitos são declarados com tamanho pré-definido, através do tamanho é utilizado um loop for para fazer a leitura dos elementos de cada posição.

PROGRAMA 2

Este exemplo cria vetores do tipo double e int, os vetores salário e idade são lidos diretamente no for. No último loop for os vetores poderão ser exibidos todos juntos, o salário é exibido com 10% de aumento.

ATENÇÃO: Antes de digitar o programa, faça a simulação no caderno com os seguintes salários e idades:

SAL= 1000, 1500, 2000

IDA= 18 , 20, 22

```
#include "iostream"
#include "math.h"
using namespace std;
int const n=3;
double sal[n];
int ida[n];
int main () { setlocale(LC_ALL,
"Portuguese-brasilian");
double x;
for (int i = 0; i < n ; i ++ ) {
    cout << "\nDigite salário e idade separados
por espaço:";
    cin >> sal[i];
    cin >> ida[i]; }
for ( int i=0; i<n; i++)
{
    x = sal[i] * 1.1;
    cout << x << " - " << ida[i] << endl;
}
return 0; }
```

TAREFA DE FIXAÇÃO / AVALIAÇÃO

Programar os exercícios na linguagem C, a saber, os enunciados A, B, C, D, E, F e G da página 74 do **Livro de Exercícios**. Se quiser, você pode fazer um único código controlado por um menu que irá executar cada exercício dentro de um void, pode também criar os vetores no início do código, fora das sub-rotinas. Pode também fazer cada exercício de forma separada. Armazene os valores em vetores com tamanho 2 pelo menos.

NOTA: Entrega obrigatória individual pelo Teams, em um arquivo DOCX ou PDF, juntamente com enunciado, programa e prints da tela de execução de cada um. A regra é descontar um ponto, no mínimo, para cada tópico não entregue.