

Programação de Computadores II

Cap. 5 – Vetores

Livro: Waldemar Celes, Renato Cerqueira, José Lucas Rangel. *Introdução a Estruturas de Dados*, Editora Campus (2004)
Slides adaptados dos originais dos profs.: Marco Antonio Casanova e Marcelo Gattass (PUC-Rio)

12/03/2014

1

Referências

Waldemar Celes, Renato Cerqueira, José Lucas Rangel, *Introdução a Estruturas de Dados*, Editora Campus (2004)

Capítulo 5 – Vetores e alocação dinâmica

12/03/2014

2

Tópicos

- Vetores

12/03/2014

3

Vetores

- Exemplo:
 - cálculo da média de um conjunto de 10 números reais

$$m = \frac{\sum x}{N}$$

- implementação
 - valores são lidos e somados
 - cálculo da média é feito a partir da soma

12/03/2014

4

```
/* Cálculo da média de n números reais */
#include <stdio.h>

int main ( void ) {
    float nota, med = 0.0f;
    int i,n;

    printf("Entre com o número de notas:");
    scanf("%d",&n);

    /* leitura do conjunto de valores e cálculo do somatório */
    for (i = 0; i < n; i++) {
        scanf("%f", &nota);
        med = med + nota;
    }

    /* cálculo da média */
    med = med / n;

    /* exibição do resultado */
    printf ("Media = %f \n", med);
    return 0;
}
```

12/03/2014

5

Alocação de várias variáveis

```
int a1,a2,a3;
```

Pilha de memória

		7020
		7014
		7012
a3	-	7008
a2	-	7004
a1	-	7000

12/03/2014

6

```
int a1, a2, a3;  
char letra;
```

Pilha de memória

letra	-	7013
a3	-	7012
a2	-	7008
a1	-	7004
	-	7000

12/03/2014

7

Vetores

- **Vetor:**
 - estrutura de dados definindo um conjunto homogêneo
 - Exemplo:
 - `v = vetor de inteiros com 10 elementos`

```
int v[10];
```

neo		
	-	144
	-	140
v	-	136
	-	132
	-	128
	-	124
	-	120
	-	116
	-	112
	108	
	104	

V

12/03/2014

Vetores: inicialização

```
int v[10]={2,3,5,7,11,13,17,19,23,29};
```

Liberalidade da linguagem:

```
int v[]={2,3,5,7,11,13,17,19,23,29};
```

		144
29		140
23		136
19		132
17		128
13		124
11		120
7		116
5		112
3		108
2		104

v

12/03/2014

Vetores: acesso aos elementos

```
int v[10]={2,3,5,7,11,13,17,19,23,29};
```

~~v[10]=31;~~

não pode!
invade a memória da
próxima variável!

v[9]	29	140
v[8]	23	136
v[7]	19	132
v[6]	17	128
v[5]	13	124
v[4]	11	120
v[3]	7	116
v[2]	5	112
v[1]	3	108
v[0]	2	104

12/03/2014

Acesso aos elementos: exemplo

```
int v[7]={1,3,5,7,9,11,13};
int i;
```

```

ou:
int v[7];
int i;

for (i=0; i<7; i++) {
    v[i]=2*i+1;
}

```

		140
		136
i		132
v[6]	13	128
v[5]	11	124
v[4]	9	120
v[3]	7	116
v[2]	5	112
v[1]	3	108
v[0]	1	104

i

12/03/2014

Vetores e Ponteiros

```
int v[7];
```

Quem é v?

v é um ponteiro de valor constante que aponta para o primeiro elemento do vetor de inteiros!

		140
		136
		132
	13	128
	11	124
	9	120
	7	116
	5	112
	3	108
v →	1	104

V →

12/03/2014

Vetores e Aritmética de Ponteiros

```
int v[7];
```

v+i aponta para v[i]

Ou seja: se i=1:

```
*(v+i)=3; ⇔ v[i]=3;
```

```
scanf("%d",&v[i]);
```

⇔

```
scanf("%d",v+i);
```

12/03/2014

v+6 →	13	140
v+5 →	11	136
v+4 →	9	132
v+3 →	7	128
v+2 →	5	124
v+1 →	3	120
v+0 →	1	116

Vetores

- Exemplo:

- cálculo da média e da variância de um conjunto de 10 números reais

$$m = \frac{\sum x}{N}, \quad v = \frac{\sum (x-m)^2}{N}$$

- implementação

- valores são lidos e armazenados em um vetor de 10 posições
- cálculos da média e da variância efetuados sobre o conjunto de valores armazenado

12/03/2014

14

```
/* Cálculo da média e da variância */
#include <stdio.h>

#define MAX 10

int main ( void ) {
    float v[MAX], med=0.0, var=0.0;
    int i,n;

    for (i = 0; i < MAX; i++)
        scanf("%f", &v[i]);

    for (i = 0; i < MAX; i++)
        med = med + v[i];
    med = med / MAX;

    for (i = 0; i < MAX; i++)
        var = var + (v[i]-med)*(v[i]-med);
    var = var / MAX;

    printf ( "Media = %f   Variância = %f  \n", med, var );
    return 0;
}
```

12/03/2014

15

Vetores

- Passagem de vetor para função:

- consiste em passar o endereço da primeira posição do vetor
- função deve ter parâmetro do tipo ponteiro para armazenar esse endereço
 - "passar um vetor para uma função" é equivalente a "passar o endereço inicial do vetor"
 - elementos do vetor não são copiados para a função
 - argumento copiado é apenas o endereço do primeiro elemento

- Exemplo:

- Na chamada à função que passa um vetor de `int`, a função deve ter um parâmetro do tipo `int *`

12/03/2014

16

```
/* Cálculo da média e da variância de n reais (segunda versão) */
#include <stdio.h>

/* Função para cálculo da média */
float media (int n, float *v)
{
    int i;
    float s = 0.0;
    for (i = 0; i < n; i++)
        s += v[i];
    return s/n;
}
```

parâmetro do tipo
ponteiro para float

12/03/2014

17

```
/* Função para cálculo da variância */
float variancia (int n, float *v, float m)
{
    int i;
    float s = 0.0;
    for (i = 0; i < n; i++)
        s += (v[i] - m) * (v[i] - m);
    return s/n;
}
```

12/03/2014

18

```
#define MAX 10

int main ( void )
{
    float v[MAX];
    float med, var;
    int i;

    /* leitura dos valores */
    for ( i = 0; i < MAX; i++ ) {
        scanf("%f", &v[i]);
    }

    med = media(MAX,v);
    var = variancia(MAX,v,med);

    printf ( "Media = %f   Variancia = %f  \n", med, var);
    return 0;
}
```

12/03/2014

19

Vetores

- Passagem de vetor para função (cont.):
 - função pode alterar os valores dos elementos do vetor pois recebe o endereço do primeiro elemento do vetor (e não os elementos propriamente ditos)
 - Exemplo:
 - função incrementando todos os elementos de uma unidade

12/03/2014

20

```
/* Incrementa elementos de um vetor */
#include <stdio.h>
void incr_vetor ( int n, int *v )
{
    int i;
    for ( i = 0; i < n; i++ )
        v[i]++;
}

int main ( void )
{
    int a[ ] = {1, 3, 5};
    incr_vetor(3, a);
    printf("%d %d %d \n", a[0], a[1], a[2]);
    return 0;
}
```

saída do programa será 2 4 6

12/03/2014

21

Exercício

- Escreva uma função que recebe dois vetores de tamanho n e um terceiro vetor de tamanho $2n$. Ao chamar a função os elementos dos vetores de tamanho n são inseridos de modo intercalado no vetor de tamanho $2n$.

12/03/2014

22

```
/* Incrementa elementos de um vetor */
#include <stdio.h>
void une_intercalado ( int n, int *u, int *v, int *z )
{
    int i;
    for ( i = 0; i < n; i++ ) {
        z[2*i] = u[i];
        z[2*i+1] = v[i];
    }
}

int main ( void )
{
    int a[ ] = {1, 3, 5};
    int b[ ] = {2, 4, 6};
    int c[6];
    int i;
    une_intercalado(3, a, b, c);
    for ( i = 0; i < 6; i++ )
        printf("%d\n", c[i]);
    return 0;
}
```

12/03/2014

23

Revisão

n++ incrementa n de uma unidade, depois de ser usado
n-- decrementa n de uma unidade, depois de ser usado

if (*expr*) { *bloco de comandos* } else { *bloco de comandos* }

for (*expr_inicial*; *expr_booleana*; *expr_de_incremento*) { *bloco de comandos* }

"passar um vetor para uma função" = "passar o endereço inicial do vetor"

exemplo 1: float variancia (int n, float* v, float m)

exemplo 2: void incr_vetor (int n, int *v)

alocação dinâmica de vetor: através da função malloc

exemplo 3: int *v;

v = (int *) malloc(10*sizeof(int));

12/03/2014

24