




UNIVERSIDADE
FUMEC

Algoritmos e Estrutura de Dados
Videoaula 03 (Parte 1/3)

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG




UNIVERSIDADE
FUMEC

Revisando a última aula!

Algoritmos e Estrutura de Dados

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

2



UNIVERSIDADE
FUMEC

Segundo Programa

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

3

Programa 02 – convDiaAno.c



```

1  /* Meu Segundo Programa */
2  #include <stdio.h>
3  #include <stdlib.h>
4  int main (void)
5  {
6      int dias;
7      float anos;
8      setbuf(stdout, NULL); /*ATENCAO!!!*/
9      printf ("Digite o número de dias: ");
10     fflush(stdin);
11     scanf ("%d",&dias);
12     anos=dias/365.25;
13     printf ("\n%d dias equivalem a %f anos.\n",dias,anos);
14     return 0;
15 } //Fim

```

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

4

Estruturas de *Decisão* e de *Repetição*



Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

5

Estruturas de decisão e repetição



- ❑ Estruturas de *decisão*:
 - ❑ if
 - ❑ if-else
 - ❑ if-else-if
 - ❑ If's aninhados
 - ❑ O comando Switch
- ❑ Estruturas de *repetição*:
 - ❑ O comando for
 - ❑ O comando while
 - ❑ O comando do-while

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

6

Funções

Funções

```
tipo_de_retorno nome_da_função (lista_de_argumentos)
{
    código_da_função
}

/* Funcao simples: Só imprime Ola! */
void mensagem ()
{
    printf ("Ola! ");
}
```

Argumentos de Funções

Argumentos de Funções



- ❑ Na **definição** da função `square()` dizemos que a função receberá um argumento *inteiro* **x**.

```
void square (int x)
```

- ❑ Quando fazemos a **chamada** à função, o inteiro **num** é passado como argumento.

```
square (num)
```

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

10

Os comandos *break* e *continue*



Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

11

Objetivos de Aprendizagem

- ❑ Ponteiros.
- ❑ Vetores e Alocação Dinâmica **Fluxo:**
 - ❑ Estrutura de decisão.
 - ❑ O comando **switch**.
 - ❑ Estrutura de repetição.
 - ❑ Comandos **break** e **continue**.
- ❑ Matrizes.



A Linguagem C

Ponteiros

Ponteiro

É uma variável.

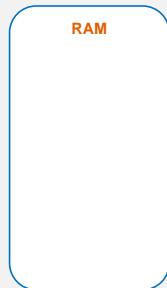
- ☐ Ou seja, espaço de memória para armazenar dados!

Ponteiros armazenam endereços de variáveis que estejam na memória.

- ☐ Essas variáveis obrigatoriamente devem possuir o mesmo tipo declarado do ponteiro.

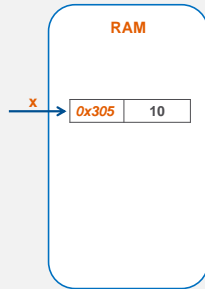
Ponteiro

- ☐ Memória RAM



Ponteiro

❑ `int x = 10`



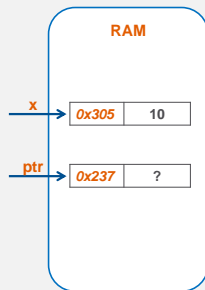
Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

16

Ponteiro

❑ `int x = 10`

❑ `int *ptr;`



Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

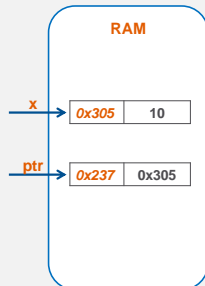
17

Ponteiro

❑ `int x = 10`

❑ `int *ptr;`

❑ `ptr = &x;`

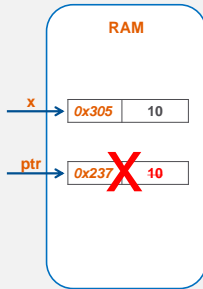


Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

18

Ponteiro

```
int x = 10;
int *ptr;
ptr = x;
```



Vamos praticar?

Exercício de Fixação - Ponteiros

Implemente um programa na linguagem C seguindo as seguintes instruções:

1. Crie as variáveis x e y para receber inteiros.
2. Atribua o valor 25 a variável x.
3. Crie três ponteiros, ptr1, ptr2 e ptr3.
4. Atribua a ptr1 o endereço de x.
5. Atribua a y, o conteúdo da variável apontada por ptr1, multiplicada por 2.
6. Atribua o endereço de y a ptr2.
7. Atribua o ponteiro ptr2 ao ponteiro ptr3.
8. Imprima o endereço de todas as variáveis (+ ponteiros).
9. Imprima o conteúdo de todas as variáveis (+ ponteiros).

No próximo bloco...

Continuaremos nossos estudos sobre a Linguagem C, agora sobre:

- ❑ Vetores.

Bons estudos!

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

22

Algoritmos e Estrutura de Dados
Videoaula 03 (Parte 2/3)

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

UNIVERSIDADE FUMEC

A Linguagem C

Vetores

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

24

Vetores

❑ Memória RAM

RAM

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

25

Vetores

❑ `int vetor[10];`

RAM

| | |
|---|-----|
| 9 | int |
| 8 | int |
| 7 | int |
| 6 | int |
| 5 | int |
| 4 | int |
| 3 | int |
| 2 | int |
| 1 | int |
| 0 | int |

vetor

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

26

Vetores

❑ `vetor[3]=10`

RAM

| | |
|---|-----|
| 9 | int |
| 8 | int |
| 7 | int |
| 6 | int |
| 5 | int |
| 4 | int |
| 3 | 10 |
| 2 | int |
| 1 | int |
| 0 | int |

vetor

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

27

Vetores

❑ `vetor[0]=5`

RAM

| | |
|---|-----|
| 9 | int |
| 8 | int |
| 7 | int |
| 6 | int |
| 5 | int |
| 4 | int |
| 3 | 10 |
| 2 | int |
| 1 | int |
| 0 | 5 |

vetor →

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

28

A Linguagem C

Alocação Dinâmica

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

29

Alocação Dinâmica

❑ Memória RAM

RAM

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

30

Alocação Dinâmica

```
int *vet;
```



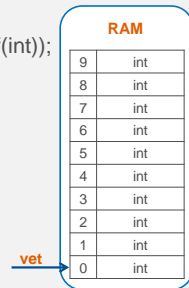
Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

31

Alocação Dinâmica

```
int *vet;
```

```
vet = (int*) malloc (10* sizeof(int));
```



Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

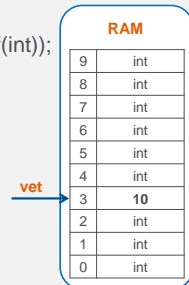
32

Alocação Dinâmica

```
int *vet;
```

```
vet = (int*) malloc (10* sizeof(int));
```

```
vet[3] = 10;
```

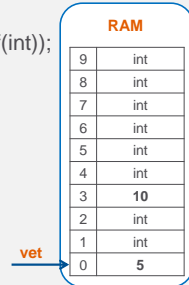


Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

33

Alocação Dinâmica

- ❑ `int *vet;`
- ❑ `vet = (int*) malloc (10* sizeof(int));`
- ❑ `vet[3] = 10;`
- ❑ `*(vet+0) = 5;`



Vamos *praticar*?

Exercício de Fixação

1. Implemente um programa na linguagem C para calcular a média aritmética entre 5 inteiros digitados pelo usuário (Utilize a função "scanf()").
2. Armazene cada inteiro em uma posição do vetor.
3. Utilize, pelo menos, uma estrutura de repetição (for ou while) em seu programa.

No próximo bloco...

Continuaremos nossos estudos sobre a Linguagem C, agora sobre:

- ❑ Matrizes.

Bons estudos!

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

37

Algoritmos e Estrutura de Dados
Videoaula 03 (Parte 3/3)

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

UNIVERSIDADE FUMEC

A Linguagem C

Matrizes

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

39

Matriz

Memória RAM



RAM

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

40

Matriz - Declaração

`int mat[3][2];`



RAM

mat

| | |
|---|-----|
| 5 | int |
| 4 | int |
| 3 | int |
| 2 | int |
| 1 | int |
| 0 | int |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

41

Matriz - Inicialização

`int mat[3][2] = {10,20,30,40,50,60};`
 OU
`int mat[3][2] = {{10,20},{30,40},{50,60}};`
 OU
`int mat[][2] = {10,20,30,40,50,60};`



RAM

mat

| | |
|---|----|
| 5 | 60 |
| 4 | 50 |
| 3 | 40 |
| 2 | 30 |
| 1 | 20 |
| 0 | 10 |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

42

Matriz - Acesso

❑ `int mat[3][2] = {10,20,30,40,50,60};`

$i \downarrow$ $j \rightarrow$

| | |
|----|----|
| 10 | 20 |
| 30 | 40 |
| 50 | 60 |

mat

RAM

| | |
|---|----|
| 5 | 60 |
| 4 | 50 |
| 3 | 40 |
| 2 | 30 |
| 1 | 20 |
| 0 | 10 |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

43

Matriz – Representada por Vetor

❑ $k = i*n + j = \text{posição (n = colunas)}$

$i=1 \downarrow$ $j=1 \rightarrow$

| | |
|---|---|
| a | b |
| c | d |
| e | f |

| | | | | | |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 |
| a | b | c | d | e | f |

mat

RAM

| | |
|---|---|
| 5 | f |
| 4 | e |
| 3 | d |
| 2 | c |
| 1 | b |
| 0 | a |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

44

Matriz – Representada por Vetor

❑ $k = i*n + j = 1*2 + 1 = 3$

$i=1 \downarrow$ $j=1 \rightarrow$

| | |
|---|---|
| a | b |
| c | d |
| e | f |

| | | | | | |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 |
| a | b | c | d | e | f |

mat

RAM

| | |
|---|---|
| 5 | f |
| 4 | e |
| 3 | d |
| 2 | c |
| 1 | b |
| 0 | a |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

45

Matriz – Representada por Vetor

$k = i*n+j = 1*2+1 = 3$

RAM

| | |
|---|---|
| 5 | f |
| 4 | e |
| 3 | d |
| 2 | c |
| 1 | b |
| 0 | a |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

Matriz – Vetor de Ponteiros

$k = i*n+j = 1*2+1 = 3$

RAM

| | |
|---|---|
| 0 | 1 |
| a | b |
| 2 | 3 |
| c | d |
| 4 | 5 |
| e | f |

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG

Vamos praticar?

Prof. Rafael Nunes – M.Sc. em Engenharia de Computação pela UFMG



Exercício de Fixação

Implemente um programa na linguagem C para **subtrair** duas matrizes seguindo as instruções abaixo:

1. Leia duas matrizes 2x2 inteiras (somente números inteiros) do usuário.
2. Envie as duas matrizes lidas para uma função.
3. Realize a subtração e imprima na tela as duas matrizes e o resultado.



Para finalizar...

**Faça os exercícios propostos
e não deixe de acessar o
*ambiente virtual.***

**Bons estudos e até a nossa
próxima videoaula!**

Referências

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