



Introdução à Programação

Eduardo Silva Lira

XLVIII Programa de Verão do IME-USP

São Paulo - SP, Jan 2019



INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
UNIVERSIDADE DE SÃO PAULO



Vamos pensar...

- É possível armazenar diversos valores numéricos?
 - Inteiro?
 - Ponto flutuante?

Arrays em C

- Vamos comparar:
 - Uma variável normal só guarda **um valor** de **um tipo**
 - Um **array** armazena **n valores** de **um tipo**
- Arrays de 1 dimensão são chamados de vetores

Sintaxe:

```
<tipo> <identificador> [<tamanho>] ;
```

Arrays em C

Sintaxe:

```
<tipo> <identificador> [<tamanho>];
```

Exemplos:

```
unsigned int idade[10]; /*10 idades*/  
double preco[25]; /*25 Preços*/  
int num[50]; /*50 números*/  
/*Para armazenar a qtde vendida em  
  *cada mes*/  
unsigned short vendaPorMes[12];
```

Arrays em C

- É um **vetor**!

```
int primos[10];
```

Cada posição são 4
bytes na memória
(se `sizeof(int) == 4`)



Arrays em C

- É um **vetor**!

```
int primos[10];
```

Cada posição são 4 bytes na memória
(se `sizeof(int) == 4`)

A última posição também armazena



0	1	2	3	4	5	6	7	8	9
2	3	5	7	11	13	17	23	29	31

Arrays

- É possível inicializar arrays?

```
int contIdade[5] = {0, 0, 0, 0, 0};
```

Arrays

- É possível inicializar arrays?

```
int contIdade[5] = {0, 0, 0, 0, 0};
```

*E se fossem muitos itens, como
inicializar?*

Arrays

- É possível inicializar arrays?
 - Veja mais dois exemplos (equivalentes):

```
char myHexa[16] = { '0', '1', '2', '3',  
                    '4', '5', '6', '7',  
                    '8', '9', 'A', 'B',  
                    'C', 'D', 'E', 'F' };
```

```
char myHexa[16] = "0123456789ABCDEF";
```

Como Ler Arrays

- Utilizar **scanf**
 - Já conhecemos os formatadores

Como Ler Arrays

- Utilizar **scanf**
 - Já conhecemos os formatadores

```
unsigned short idade[5];
```

```
/*le a primeira idade*/  
scanf("%u", &idade[0]);
```

Como Ler Arrays

- Utilizar **scanf**

```
unsigned short idade[5];
```

```
/*le a segunda idade*/  
scanf("%u", &idade[1]);
```



Observem!

Como Ler Arrays

Na parametrização, **idade[2]** e **nome[5]** retornam o **valor** da posição enquanto que **&idade[4]** e **&nome[2]** retornam o **endereço** da posição

- Utilizar **scanf**

```
unsigned short idade[5];
```

```
/*le a segunda idade*/  
scanf("%u", &idade[1]);
```

Observem!

Quando lemos uma string, vamos guardar no **vetor todo**, não em uma **posição**.

```
scanf("%s", nome);
```

Arrays - Exemplos

- Faça um programa para:
 - Ler 4 (quatro) notas [0.0, 10.0]
 - Calcular a **média**
 - Exibir media e notas

Arrays - Exemplos - Resolução

Exemplo:

```
#include<stdio.h>
/*Constante para definir a qtde de notas*/
#define N_NOTAS 4
int main(){
    double nota[N_NOTAS];
    double media = 0;
    int i;

    for(i = 0; i < N_NOTAS; ++i){
        printf("Digite a nota %d: ", (i+1));
        scanf("%lf", &nota[i]);/*Ler a i-esima nota*/
        media = media + nota[i];/*Acumular as notas*/
    }
    media = media / N_NOTAS;/*Calcular a media*/

    /*Exibir as notas*/
    for(i = 0; i < N_NOTAS; ++i){
        printf("Nota %d = %5.2f\t", (i+1), nota[i]);
    }

    printf("\nMedia = %5.2f\n", media);/*Exibir a media*/

    return 0;
}
```

Arrays - Exemplos

- Crie um programa para ler 10 preços (R\$)
- Imprima na ordem **inversa** a digitada

Dica! Validar utilizando **break**

```
#include<stdio.h>

#define TRUE 1
#define MAX_PRICE 50.00

int main(){
    double preco;

    while(TRUE){
        printf("Digite o preco R$ [0.0 ate %.2f]: " , MAX_PRICE);
        scanf("%lf", &preco);
        if (preco < 0.0 || preco > MAX_PRICE){
            printf("Valor invalido!\n");
        }else{
            break;
        }
    }

    printf("Preco informado: R$%.2f. \n", preco);
    return 0;
}
```

Arrays - Exemplos

- Crie um programa para ler 10 preços (R\$)
- Imprima na ordem **inversa** a digitada

Altere para trabalhar com **n** preços.

Arrays - Exemplos

- Tentando descobrir se um **dado** era **viciado**, um dono de cassino honesto (ha! ha! ha! ha!) o lançou ***n*** vezes. Dados os ***n*** resultados dos lançamentos, determinar o **número de ocorrências de cada face**.

Armazenar todos os lançamentos!!!!

Opção 1: O programa lê até o usuário digitar **0** (zero) para a **face** (resultado do lançamento).

Opção 2: Pergunte ao usuário quantas vezes ele irá jogar o dado.

O valor de *n* é no máximo 1000.

FAZER USANDO FOR



Dúvidas?