

Fundamentos de Programação 1

Linguagem C

Estrutura de Dados Homogêneas “**Vetores**”

Slides 03 – ‘Vetores’

Prof. SIMÃO

Exercício 1.

Elabore um programa que, dada a idade de um nadador (entrada do algoritmo), classifique-o de acordo com as seguintes categorias:

- infantil A = 5 – 7 anos;
- infantil B = 8 – 10 anos;
- juvenil A = 11 – 13 anos;
- juvenil B = 14 – 17 anos;
- adulto = maiores ou igual a 18 anos.

Exercício – 2.

Refaça o programa anterior para o caso de 50 nadadores, definindo a média de idade dos nadadores, a idade do mais velho e a idade do mais jovem.

Uma solução para o exercício 2.

```
void main ( )
{
    int  cont;
    float idade, soma, media, maior, menor;

    soma = 0.0;
    maior = 0.0;
    menor = 200.0;

    for ( cont = 0; cont < 50; cont = cont + 1 )
    {
        printf ( "Digite a idade do nadador: " );
        scanf ( "%f", &idade );

        if ( ( idade < 5 ) || ( idade > 99 ) )
        {
            printf ( " Idade inválida. " );
        }
        else
        {
            soma = soma + idade;

            if ( idade > maior )
            {
                maior = idade;
            }

            if ( idade < menor )
            {
                menor = idade;
            }
        }
    }
}
```

```
        if ( ( idade >= 5 ) && ( idade < 8 ) )
        {
            printf ( " Idade na categoria infantil A. " );
        }
        else
        {
            // ...
        }
    }
}
media = soma / 50.0;

printf ( " A média de idade é %f: ", media );
printf ( " A maior idade é %f:", maior );
printf ( " A menor idade é %f.", menor );
}
```

Retomando o exercício 2.

Refaça o programa anterior dizendo agora quantos nadadores têm idade maior que a média.

Estrutura de Dados Homogênea

“vetores”

Um vetor computacional é um “variável composta” capaz de armazenar uma quantidade determinada de valores.

Um vetor computacional é inspirada no vetor matemático, que também é capaz de armazenar um conjunto de valores.

Em geral, um vetor é uma matriz de uma única linha ou uma única coluna.

1	3	7	0	1	0	1	9	9
---	---	---	---	---	---	---	---	---

Declaração de Vetores em linguagem C.

<tipo-básico> NOME_VETOR [número];

Onde :

número representa a capacidade de armazenamento do vetor;

< tipo-básico > - representa qualquer um dos tipos básicos

Exemplo 1

```
void main()
{

    float vet_nums [5];
    // 5 posições, de 0 à 4... (sempre começa por zero em C).

    float media = 0.0, quant = 0.0;

    vet_nums[0] = 9.0;
    vet_nums[1] = 8.0;
    vet_nums[2] = 7.0;
    vet_nums[3] = 7.0;
    vet_nums[4] = 7.0;

    quant = vet_nums[0] + vet_nums[1] + vet_nums[2] + vet_nums[3] + vet_nums[4];

    media = quant / 5.0;

    printf (“ A média dos 5 números é: %f.”, media);

}
```

Exemplo 2

```
void main ( )
{
    float vet_nums[5];
    float media = 0.0, quant = 0.0;

    printf ( "Digite o 1º número: \n" );
    scanf ( "%f", &vet_nums[0] );

    printf ( "Digite o 2º número: \n" );
    scanf ( "%f", &vet_nums[1] );

    printf ( "Digite o 3º número: \n" );
    scanf ( "%f", &vet_nums[2] );

    printf ( "Digite o 4º número: \n" );
    scanf ( "%f", &vet_nums[3] );

    printf ( "Digite o 5º número: \n" );
    scanf ( "%f", &vet_nums[4] );

    quant = vet_nums[0] + vet_nums[1] + vet_nums[2] + vet_nums[3] + vet_nums[4];
    media = quant / 5.0;

    printf ( " A média dos 5 números é: %f.", media);
}
```

Exemplo 3

```
void main () // 'Idade Nadadores ...'.
{

    float vet_nums[5];
    float media = 0.0, quant = 0.0;
    int cont;

    for ( cont = 0; cont < 5; cont = cont + 1 )
    {
        printf ( "Digite o %i .o número:", cont+1 );
        scanf ( "%f", &vet_nums[ cont ] );

        quant = quant + vet_nums [ cont ] ;
    }

    media = quant / 5.0;

    printf ( " A média dos 5 números é: %f.", media);
}
```

Retomando o exercício

a) Elabore um programa que, dada a idade de um nadador (entrada do algoritmo), classifique-o de acordo com as seguintes categorias:

- infantil A = 5 – 7 anos;
- infantil B = 8 – 10 anos;
- juvenil A = 11 – 13 anos;
- juvenil B = 14 – 17 anos;
- adulto = maiores ou igual a 18 anos.

b) Refaça o programa anterior para o caso de 50 nadadores, definindo a média de idade dos nadadores, a idade do mais velho e a idade do mais jovem.

c) Refaça o algoritmo anterior dizendo agora quantos nadadores têm idade maior que a média.

Eis o algoritmo do programa pedido.

Algoritmo 'Idade Nadadores ...'.

Início

Real vetor_idades[50];

Inteiro cont, quant;

Real idade, soma, média, maior, menor;
soma ← 0; maior ← 0; menor ← 200;

para cont de 0 até 49 passo 1 faça

Imprima ("Digite a idade do nadador: ");

Leia (idade);

se ((idade >= 5) e (idade <= 100)) então

 // ...

 soma ← soma + idade;

se (idade < menor) então

 menor ← idade;

fim-se

se (idade > maior) então

 maior ← idade;

fim-se

vetor_idades[cont] ← idade;

senão

Imprima (" Idade inválida");

 cont ← cont -1;

fim-se

fim-para

média ← soma / 50.0;

quant ← 0;

para cont de 0 até 49 passo 1 faça

se (vetor_idades [cont] > média) então

 quant ← quant + 1;

fim-se

fim-para

Imprima (" A média de idade é %f: ", media);

Imprima (" A maior idade é %f:", maior);

Imprima (" A menor idade é %f.", menor);

Imprima (" A quantidade de idades acima da média
é %i: ", quant);

Fim.

Eis o programa pedido.

```
// Programa 'Idade Nadadores ...'.
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
void main()
```

```
{
```

```
    float vetor_idades [50];
```

```
    int quant = 0;
```

```
    float idade, soma, media, maior, menor;
```

```
    soma = 0.0; maior = 0.0; menor = 200.0;
```

```
    int cont;
```

```
    for ( cont = 0; cont < 50; cont = cont + 1 )
```

```
    {
```

```
        printf ( "Digite a idade do nadador: \n" );
```

```
        scanf ( "%f", &idade );
```

```
        if ( ( idade >= 5 ) && ( idade <= 100 ) )
```

```
        {
```

```
            // ...
```

```
            soma = soma + idade;
```

```
            if ( idade < menor )
```

```
            {
```

```
                menor = idade;
```

```
            }
```

```
            if ( idade > maior )
```

```
            {
```

```
                maior = idade;
```

```
            }
```

```
            vetor_idades [cont] = idade;
```

```
        }
```

```
    else
```

```
    {
```

```
        printf ( " Idade inválida \n" );
```

```
        cont = cont -1;
```

```
    }
```

```
}
```

```
media = soma / 50.0;
```

```
quant = 0;
```

```
for (cont = 0; cont < 50; cont = cont + 1)
```

```
{
```

```
    if ( vetor_idades [cont] > media )
```

```
    {
```

```
        quant = quant + 1;
```

```
    }
```

```
}
```

```
printf ( " A média de idade é %f: \n", media);
```

```
printf ( " A maior idade é %f: \n", maior );
```

```
printf ( " A menor idade é %f. \n", menor );
```

```
printf ( " A quantidade de idades acima da média é %i: \n", quant);
```

```
system ( "pause" );
```

```
}
```

Exercício

- Elabore um programa que recebe 100 valores e os armazene em um vetor. O programa deve ainda ter um outro vetor de 50 posições que recebe a soma de dois valores do primeiro vetor na seguinte ordem:
 - $\text{vet2}[0] \leftarrow \text{vet1}[0] + \text{vet1}[1]$,
 - $\text{vet2}[1] \leftarrow \text{vet1}[2] + \text{vet1}[3]$
 - e assim por diante.

Exercícios Complementários

a) Dado um vetor VET declarado como: Inteiro Vet[5];

- preenchê-lo (em todas as posições) com o valor 30.
- preenchê-lo com os inteiros 1, 2, 3...
- Preenchê-lo com 1 se a posição é par e com 0 se a posição é ímpar.

b) Escreva em programa que gere e imprima um vetor Y obtido pela inversão da ordem de 200 valores numéricos lidos do teclado.

c) Fazer um programa que:

- Leia dois vetores contendo, cada um, 25 elementos numéricos.
- Intercale os elementos destes dois conjuntos formando um novo vetor de 50 elementos.
- Imprima o novo vetor obtido.