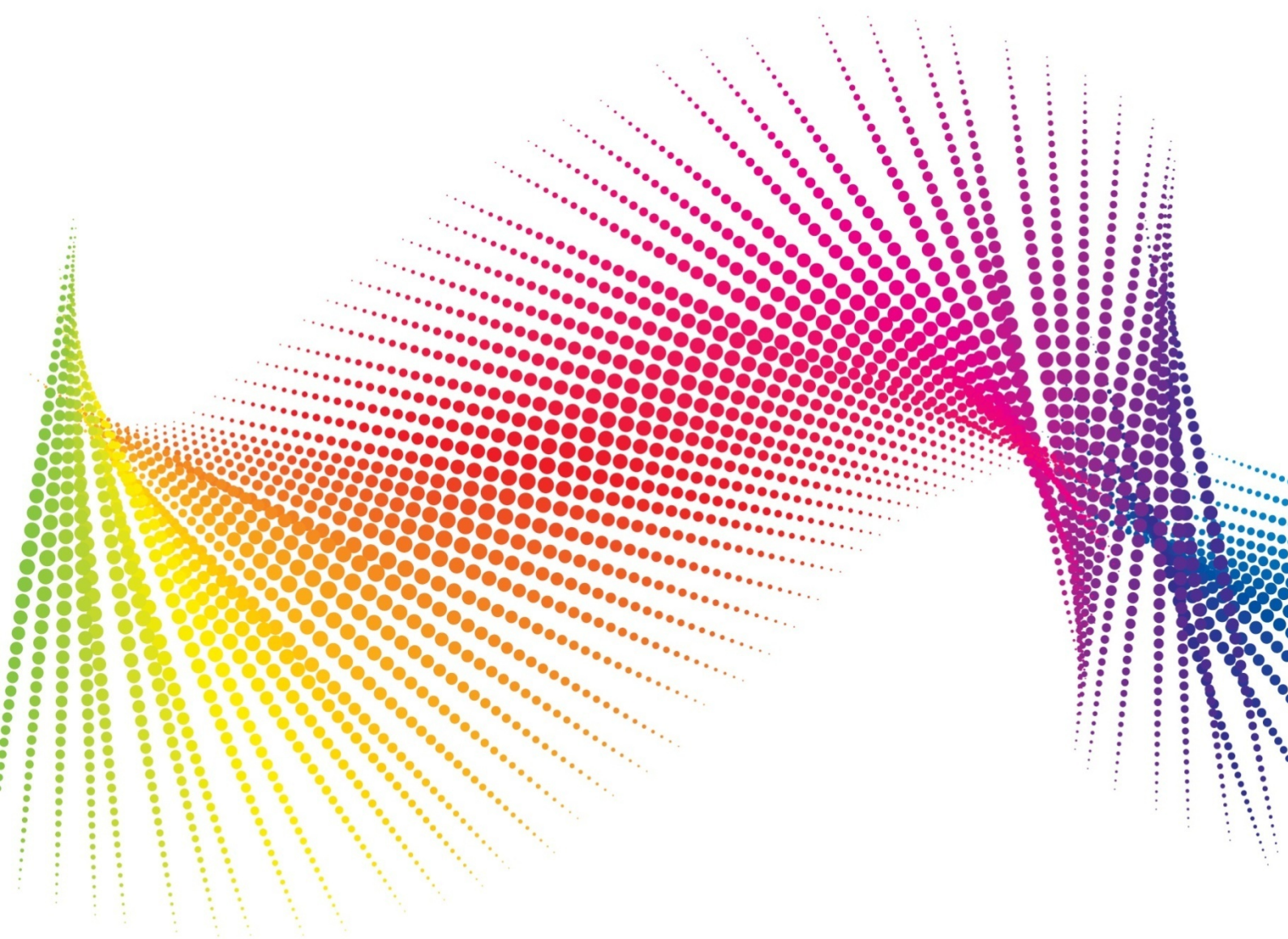


Estrutura de Dados

Aula 10



Este material é parte integrante da disciplina oferecida pela UNINOVE.

O acesso às atividades, conteúdos multimídia e interativo, encontros virtuais, fóruns de discussão e a comunicação com o professor devem ser feitos diretamente no ambiente virtual de aprendizagem UNINOVE.

Uso consciente do papel.

Cause boa impressão, imprima menos.

Aula 10: Funções

Objetivo: Continuar estudando as funções na linguagem C, agora, aprendendo como passar um vetor como parâmetro.

Passagem de vetores como parâmetros para uma função

Os vetores podem ser passados como parâmetros em funções. Mas, diferentemente das variáveis de tipos básicos em que existe a passagem por valor ou referência, **a passagem de vetores para funções é sempre por referência.**

Importante: funções não podem retornar vetores.

Quando vamos passar um vetor como argumento de uma função, podemos declarar a função de três maneiras diferentes, mas equivalentes. Vejamos o exemplo a seguir para sabermos quais são estas três maneiras.

Exemplo

Seja o vetor:

```
int vet [50];
```

e suponha que queiramos passá-la como argumento de uma função `func()`.

Podemos declarar `func()` das três maneiras seguintes:

```
void func (int vet[50]);  
void func (int vet[]);  
void func (int *vet);
```

Nos três casos, teremos dentro de `func()` um `int*` (ponteiro para inteiro) chamado `vet`. Ao passarmos um vetor para uma função, na realidade, estamos passando um ponteiro, em que é armazenado o endereço do primeiro elemento do

vetor, ou `vet[0]`. Isto significa que não é feita uma cópia, elemento a elemento, do vetor, para que possamos alterar o valor dos elementos do vetor dentro da função, ou seja, configura-se uma passagem por referência.

Mas existe, ainda, outra maneira de passar um vetor como parâmetro de uma função, que leva em conta que apenas o endereço do vetor é passado. Nesse caso, são passados 2 parâmetros: o primeiro é declarado como um vetor sem dimensão e o segundo é o tamanho do vetor. Ou seja, o tipo do vetor é declarado na lista de parâmetros da função, mas o seu tamanho não.

A informação de tamanho de vetor é passada em outro parâmetro, o que é perfeitamente possível porque a função somente precisa receber o endereço onde se encontra a primeira posição do vetor. Além disso, a linguagem C não confere limites de vetores e, portanto, a função, sabendo o endereço inicial do vetor, tem como descobrir o final do vetor.

Exemplo de programa que passa um vetor como parâmetro

```
//VetorcomoParametro.c: ilustra passagem de vetor para funcao

#include<stdio.h>
#include<conio.h>
#include <time.h>

#define MAX 10

float media (int a[], int tamanho)
{ int i;
  float soma;
  soma = 0;
  for (i=0;i<tamanho;i++)
    soma += a[i];
  return (soma/tamanho);
}

int main()
{ int i;
  int vetor[MAX];
  float m;

  printf("\nGera um vetor randomico e calcula a media ");
```

```

printf("passando o vetor como parametro para uma funcao\n");
printf("\n\nVetor de numeros randomicos gerados: ");
srand(time(NULL)); //gera um valor aleatorio inicial
for (i=0;i<MAX;i++)
{ vetor[i] = rand() % 100; //gera valores entre 0 e 99
  printf("\n vetor[%i] = %i", i,vetor[i]);
}
m = media (vetor,MAX);
printf("\n\n Media dos valores do vetor = %.2f",m);

printf("\n\n Fim do programa");
getch();
return 0;
}

```

Uma saída deste programa, que você mesmo pode e deve testar, caro aluno, seria:

```

C:\ELISA\Uninove\EaD\Lógica de Programação\C\Vetor como Parametro.exe
Gera um vetor randomico e calcula a media passando o vetor como parametro para u
ma funcao
Vetor de numeros randomicos gerados:
vetor[0] = 21
vetor[1] = 61
vetor[2] = 23
vetor[3] = 39
vetor[4] = 98
vetor[5] = 83
vetor[6] = 80
vetor[7] = 1
vetor[8] = 97
vetor[9] = 29
Media dos valores do vetor = 53.20
Fin do programa_

```

Note, no código, que o comando `srand(time(NULL))`; é que faz com que, cada vez que o programa seja executado, novos valores sejam gerados. Outro aspecto a ser notado neste código é que não foi declarado o protótipo da função, já que seu código foi declarado antes da função `main()`.

Exemplo de programa que passa um vetor como parâmetro usando ponteiro

Agora, caro aluno, vamos modificar o programa anterior e criar uma função que gere os valores randômicos do vetor e modificar a função `media`. Além disso,

vamos passar o vetor como parâmetro para as funções usando um ponteiro para vetor.

```
//VetorcomoPonteiro.c: ilustra passagem de vetor para funcao
```

```
#include<stdio.h>
#include<conio.h>
#include <time.h>
#define MAX 10

float media (int *a, int tamanho);

void geraVetor (int *vet);

int main()
{ int vetor[MAX];
  float m;

  printf("\nGera um vetor randomico e calcula a media ");
  printf("passando o vetor como ponteiro para uma funcao\n");

  printf("\n\nVetor de numeros randomicos gerados: ");
  geraVetor(vetor);
  m = media (vetor,MAX);
  printf("\n\n Media dos valores do vetor = %.2f",m);

  printf("\n\n Fim do programa");
  getch();
  return 0;
}

void geraVetor (int *vet)
{ int i;
  srand(time(NULL)); //gera um valor aleatorio inicial
  for (i=0;i<MAX;i++)
  { vet[i] = rand() % 100; //gera valores entre 0 e 99
    printf("\n vetor[%i] = %i", i,vet[i]);
  }
}

float media (int *a, int tamanho)
{ int i;
  float soma;
```

```
soma = 0;  
for (i=0;i<tamanho;i++)  
    soma += a[i];  
return (soma/tamanho);  
}
```

Agora, caro aluno, vamos praticar resolvendo os exercícios propostos. **Leia a lista, resolva os exercícios e verifique seu conhecimento.** Caso fique alguma dúvida, leve a questão ao Fórum e divida com seus colegas e professor.

REFERÊNCIAS

CRUZ, Adriano J. O. *Funções*. Disponível em: <<http://equipe.nce.ufrj.br/adriano/c/apostila/funcao.htm>>. Acesso em: 27 mai. 2012.

MIZRAHI, V.V. *Treinamento em linguagem C*. Módulo 1. 2 ed. São Paulo: Makron Books, 2006.

_____. *Treinamento em linguagem C*. Módulo 2. 2 ed. São Paulo: Makron Books, 2006.

SCHILDT, H. *C Completo e total*. 3. ed. São Paulo: Makron Books, 1997.