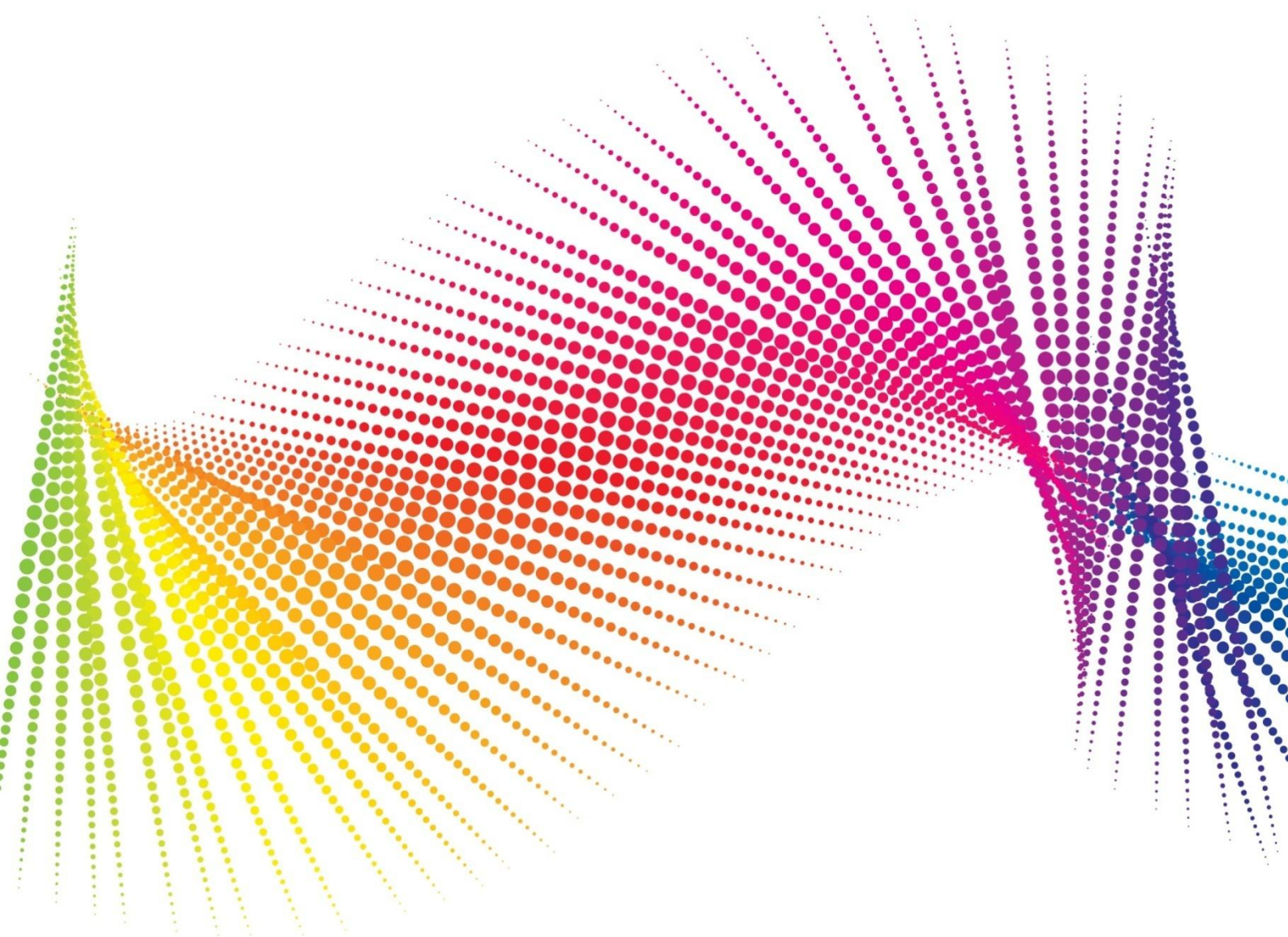


Estrutura de Dados

Aula 09



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 9: Funções

Objetivo: Continuar estudando as funções na linguagem C, com foco na passagem de parâmetros por valor ou referência.

Passagem de parâmetros para uma função: por valor e por referência

Há duas maneiras de se passar argumentos ou parâmetros para funções:

Passagem por valor:

- Esta é a maneira usual.
- O valor é copiado do argumento para o parâmetro formal da função.
- Quaisquer alterações feitas a estes parâmetros dentro da função não irão afetar as variáveis usadas como argumentos para chamá-la.

Passagem por referência:

- O que é passado como argumento no parâmetro formal é o endereço da variável.
- Dentro da função, o argumento real utilizado na chamada é acessado através do seu endereço, sendo, assim, alterado.

Exemplo:

// Declaração de um protótipo de função que recebe valores por referência

```
void func(int *i, int *j);
```

// Chamada à função, passando valores por referência

```
int i1, i2;  
func(&i1, &i2);
```

Exemplo de um programa que usa passagem de parâmetro por referência

```
/* Referencia.c: Ilustra a passagem de argumentos por
referencia em funcoes */

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

void trocal(int m, int n);
void troca2(int *m, int *n);

int main()
{int i1, i2;
 printf("Forneca o valor do primeiro inteiro\n");
 scanf("%i", &i1);
 printf("Forneca o valor do segundo inteiro\n");
 scanf("%i", &i2);

 printf("Temos i1=%d e i2=%d\n", i1,i2);
 printf("\nVamos troca-los de lugar usando a funcao trocal,
        passando-os POR VALOR a funcao.\n");
 trocal(i1,i2);

 printf("\nDepois da troca, de volta a funcao main, temos
        i1=%d e i2=%d\n", i1,i2);
 printf("\nVamos troca-los de lugar usando a funcao troca2,
        passando-os POR REFERENCIA a funcao.\n");
 troca2(&i1,&i2);

 printf("Depois da troca, de volta a funcao main, temos i1=%d
        e i2=%d\n", i1,i2);
 printf("\nFim do programa");
 getch();
 return 0;
}

void trocal(int i1, int i2)
{ int tmp;
  tmp = i1;
  i1 = i2;
  i2 = tmp;
  printf("Depois da troca, dentro da funcao trocal, temos
```

```

        i1=%d e i2=%d\n", i1,i2);
    }

void troca2(int *i1, int *i2)
{int tmp;
    tmp = *i1;
    *i1 = *i2;
    *i2 = tmp;
    printf("Depois da troca, dentro da funcao trocal, temos
        i1=%d e i2=%d\n", *i1,*i2);
}

```

Caro aluno, como você mesmo pode comprovar implementando e testando o código, a saída deste programa ao ser executado é a seguinte:

O que podemos observar é que:

- Quando as variáveis `i1` e `i2` são passados por valor à função `troca1`, elas não são alteradas na função `main()`.
- Quando as variáveis `i1` e `i2` são passados por referência à função `troca1`, elas são alteradas na função `main()`.

Caro aluno, acesse agora ao AVA e faça os exercícios de complete o espaço em branco, testando seus conhecimentos sobre funções!

Vetores de Ponteiros

Da mesma forma como podemos declarar vetores de qualquer tipo de variável, também podemos declarar **vetores de ponteiros** para qualquer tipo de variável.

Exemplo:

```

int i=5, j, *vet[5];

/* Declara dois inteiros e um vetor de ponteiros para inteiros */
/* O nome vet é um ponteiro para o vetor de ponteiros para inteiros */

vet[1] = &i;

```

/* Atribui o endereço de uma variável inteira para o elemento 1 do vetor vet */

```
j = *vet[1];
```

/* Atribui o valor do conteúdo da posição do endereço armazenado em vet[1] para j */

Exemplo de programa que usa vetores de ponteiros

//VetoresdePonteiros.c: Ilustra o uso de vetores de ponteiros

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int i;
    char *poema1[]=
    {
        "          HIPOTese",
        "E se Deus eh canhoto",
        "e criou com a mao esquerda?",
        "Isso explica, talvez, as coisas deste mundo.",
        "          Carlos Drummond de Andrade"
    };

    char *poema2[]=
    {"A vida eh a arte do encontro,\n",
    "embora haja tantos desencontros pela vida...\n",
    "          Vinicius de Moraes"
    };

    puts("Um pequeno poema:\n");
    for(i=0; i<5; i++)
        puts(poema1[i]);

    puts("\nAgora outro poema:\n");
    for(i=0; i<2; i++)
    {
        while(*poema2[i]!='\0')
            printf("%c",*(poema2[i]++));
    }
    printf("\nFim do programa");
    getch();
}
```

```
return 0;  
}
```

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

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.