

Aula 5 – Exercícios

Ponteiros

Caro aluno,
Se você for recortar as soluções dos programas deste arquivo e colá-las no compilador DevC++, **tome cuidado pois as “ ” e o sinal de – precisam ser substituídos, senão vai dar erro!!!**
Outra dica: não use acentos nas palavras, pois não funcionam nos programas...

Lista de exercícios da aula 5

Vamos praticar um pouco mais usando ponteiros. A minha forte recomendação, novamente, é que você faça os exercícios sozinho, consultando os exemplos dados para ajudá-lo. Somente depois da sua tentativa é que você deve conferir as respostas. Boa sorte!

1)Tente primeiro fazer sozinho e depois verifique no computador. Quais saídas serão apresentadas pelo programa a seguir?

```
/* Ponteiro.c: Exemplifica acesso direto e indireção*/
```

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

```
#include <conio.h>
```

```
int dado=2, *apdado;
```

```
int main( )
```

```
{
```

```
    apdado=&dado; // faz apdado apontar para dado
```

```
    printf("\nValor de dado= %i ", dado);
```

```
printf("\nValor de apdado= %p ", apdado);  
printf("\nValor de *apdado: %i ", *apdado);  
printf("\nValor de &dado: %p ", &dado);  
  
getch();  
return 0;  
}
```

2)Quais as saídas do programa a seguir? Tente primeiro descobrir e depois verifique no computador o resultado.

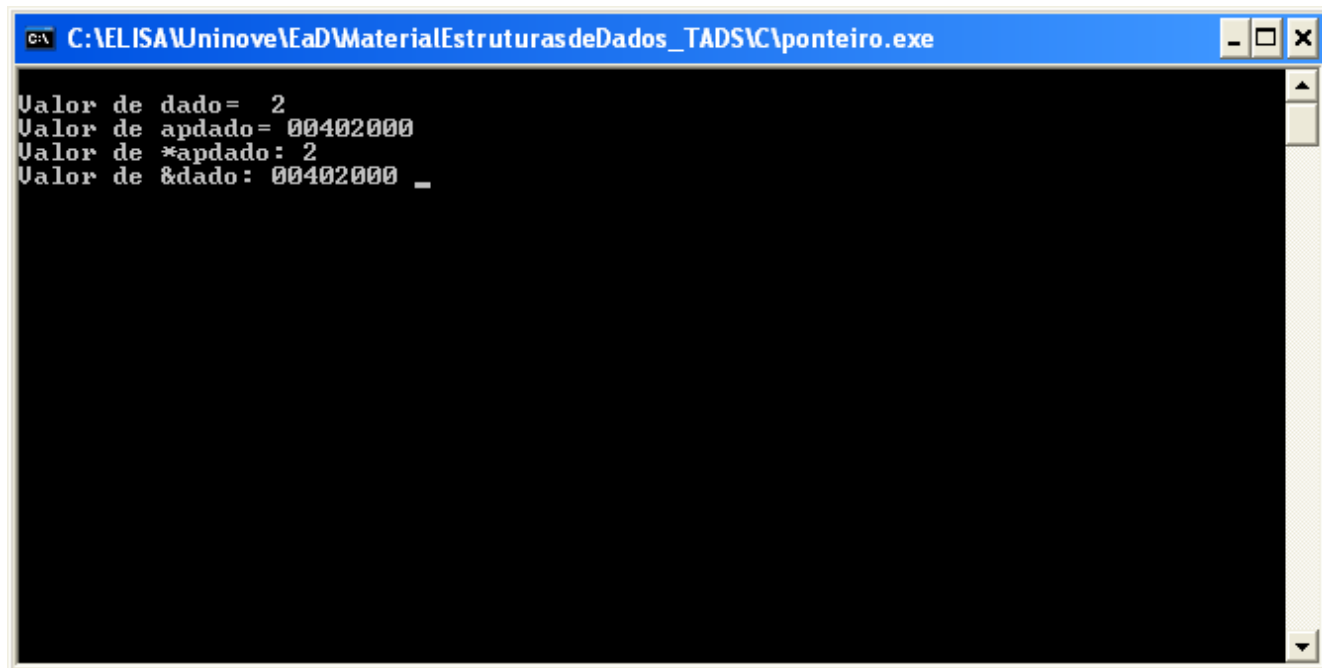
```
//ponteirosFloat.c: ilustra o uso de ponteiros  
#include <stdio.h>  
#include <conio.h>  
  
int main ()  
{ float num, *p;  
  
printf("\nDigite um numero real: ");  
scanf("%f", &num);  
p=&num; /* coloca em p o endereco de num */  
printf ("Endereco para onde o ponteiro p aponta: %p\n",p);  
printf ("Valor da variavel num apontada pelo ponteiro p: %.2f\n",*p);  
  
printf("\nDigite outro numero real: ");  
scanf("%f", &num);  
printf ("Valor da variavel num apontada por ponteiro p: %.2f \n",*p);  
  
*p=99.9;  
printf ("\nResultado apos *p=99.9; \nnum=%.1f *p=%.1f \n",num, *p);  
printf ("Endereco para onde o ponteiro p aponta: %p\n",p);  
  
printf("\n\nFim do programa");
```

```
    getch();  
    return 0;  
}
```

3) Faça um programa que declare e leia 2 números, um int e outro float, e 2 apontadores, um para int e outro para float. Depois leia os 2 números e apresente na tela seus valores e seus endereços de memória. Em seguida, multiplique o conteúdo dos números por 2 através dos apontadores declarados e apresente na tela seus valores e seus endereços de memória.

Respostas dos exercícios

1)

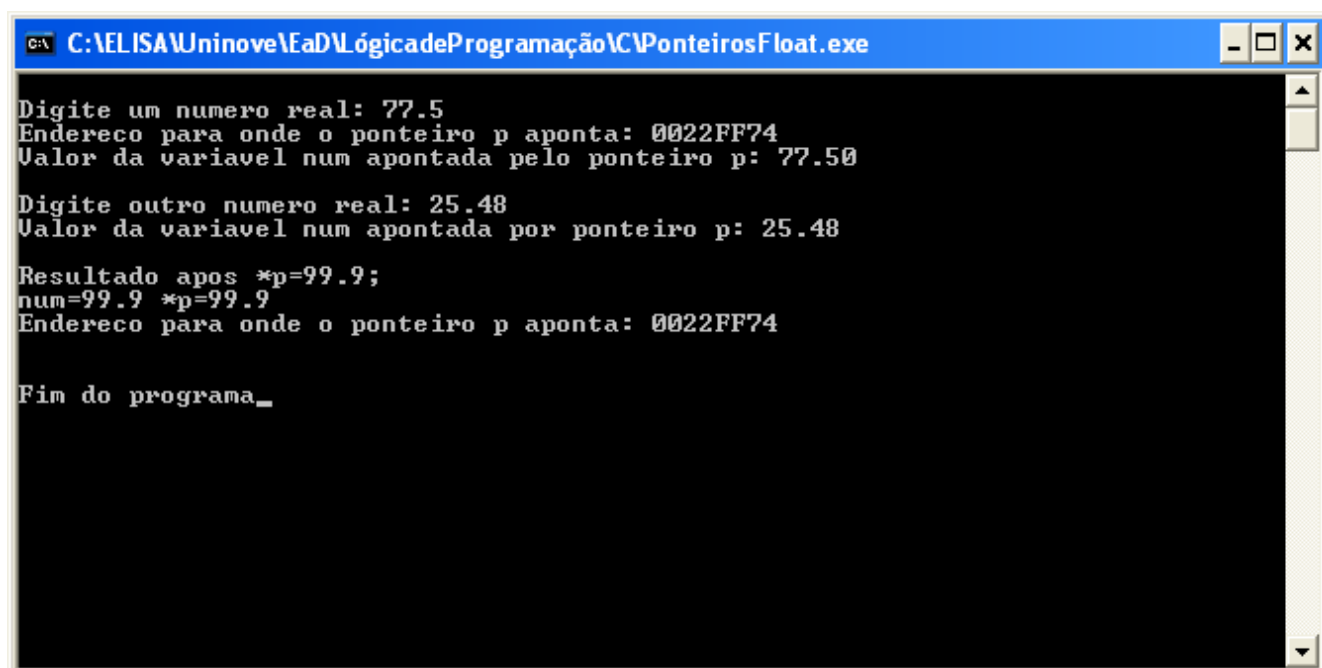


```
C:\VELISA\Uninove\EaD\MaterialEstruturasdeDados_TADSVC\ponteiro.exe

Valor de dado= 2
Valor de apdado= 00402000
Valor de *apdado: 2
Valor de &dado: 00402000 _
```

2) A execução do programa vai depender dos dados que você fornecer, caro aluno.

Veja estes resultados:



```
C:\VELISA\Uninove\EaD\Lógica de Programação\CPonteirosFloat.exe

Digite um numero real: 77.5
Endereco para onde o ponteiro p aponta: 0022FF74
Valor da variavel num apontada pelo ponteiro p: 77.50

Digite outro numero real: 25.48
Valor da variavel num apontada por ponteiro p: 25.48

Resultado apos *p=99.9;
num=99.9 *p=99.9
Endereco para onde o ponteiro p aponta: 0022FF74

Fim do programa_
```

3)

/* PonteirosIntFloat.c: 3) Faça um programa que declare e leia 2 números, um int e outro float e 2 apontadores, um para int e outro para float. Depois leia os 2 números e apresente na tela seus valores e seus endereços de memória. Em seguida, multiplique o conteúdo dos números por 2 através dos apontadores declarados e apresente na tela seus valores e seus endereços de memória.*/

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

```
int main ()
{ float f, *pf;
  int i, *pi;
```

```
printf("\nDigite um numero real: ");
scanf("%f", &f);
pf=&f; /* coloca em pf o endereco de f */
printf ("\nValor de f=%.2f endereco de f=%p endereco de pf=%p ",
        f, &f, pf);
```

```
*pf = f * 2;
printf ("\nValor de f=%.2f valor armazenado em pf apos *pf = f*2 ... =%.2f",
        f, *pf);
```

```
printf("\n\n=====\\n\\n");
```

```
printf("\nDigite um numero inteiro: ");
scanf("%i", &i);
pi=&i; /* coloca em pi o endereco de i */
printf ("\nValor de i=%i endereco de i=%p endereco de pi=%p ",
        i, &i, pi);
```

```
*pi = i * 2;  
printf ("\nValor de i=%i valor armazenado em pi apos *pi= i * 2 ... =%i ",  
        i, *pi);  
  
printf("\n\n\nFim do programa");  
getch();  
return 0;  
}
```