

Programação II

Passagem de Parâmetros

Jocélio Passos
joceliopassos@bol.com.br

tr.p2/primeira.página

Passagem de Parâmetros - Conceito

- C Comp Total, Cap 6, 139 -142
- Sebesta 323-334
- Passagem de parâmetros
 - ◆ Por valor : é feita uma cópia das variáveis originais que permanecem inalteradas.
 - ◆ Por referência : são passados os endereços de memória destas variáveis e qualquer alteração feita nessas variáveis no interior da função/procedimento refletirá na variável originalmente passada.

22/07/2007

2

tr.p2/primeira.página

Passagem de Parâmetros - Um Exemplo

- Faça um programa que leia dois números. Depois deve-se chamar um procedimento que troque o conteúdo destas duas variáveis e mostre na tela.

22/07/2007

3

tr.p2/primeira.página

Passagem de Parâmetros - Valor

- #include <stdio.h>
- #include <stdlib.h>
- #include <conio.h>
- void Trocar (int v1, int v2);
- main()
 - { int n1, n2;
 - printf ("\n Digite o valor de n1 : ");
 - scanf ("%d",&n1);
 - printf ("\n Digite o valor de n2 : ");
 - scanf ("%d",&n2);

22/07/2007

4

tr.p2/primeira.página

Passagem de Parâmetros - Valor

- printf ("\nAntes de trocar: ");
- printf ("\n n1 = %d e n2 = %d ",n1,n2);
- Trocar (n1, n2); // Por Valor
- printf ("\nDepois de trocar: ");
- printf ("\n n1 = %d e n2 = %d ",n1,n2);
- getch();
- }

22/07/2007

5

tr.p2/primeira.página

Passagem de Parâmetros - Valor

- Main ()
- {...
- Trocar (n1, n2);
- ...
- }
- void Trocar (int v1, int v2)
- { }

É feita uma cópia das variáveis originais que permanecem inalteradas.



22/07/2007

6

tr.p2/primeira.página

Passagem de Parâmetros - Valor

```
■ void Trocar (int v1, int v2)
■ { int tmp;
■   tmp = v1;
■   v1 = v2;
■   v2 = tmp;}
```

22/07/2007

7

tr.pt/primeira.página



Passagem de Parâmetros - Referência

```
■ #include <stdio.h>
■ #include <stdlib.h>
■ #include <conio.h>
■ void Trocar (int *pv1, int *pv2);
■ main()
■ { int n1, n2;
■   printf ("\n Digite o valor de n1 : ");
■   scanf ("%d",&n1);
■   printf ("\n Digite o valor de n2 : ");
■   scanf ("%d",&n2);
```

22/07/2007

8

tr.pt/primeira.página



Passagem de Parâmetros - Referência

```
■ printf ("\nAntes de trocar: ");
■ printf ("\n n1 = %d e n2 = %d ",n1,n2);

■ Trocar (&n1, &n2); // Por Referência

■ printf ("\nDepois de trocar: ");
■ printf ("\n n1 = %d e n2 = %d ",n1,n2);
■ getch(); }
```

22/07/2007

9

tr.pt/primeira.página



Passagem de Parâmetros - Referência

```
■ void Trocar (int *pv1, int *pv2)
■ { int tmp;
■   tmp = *pv1;
■   *pv1 = *pv2;
■   *pv2 = tmp;}
```

22/07/2007

10

tr.pt/primeira.página



Passagem de Parâmetros - Valor

```
■ Main ()
■ {...
■ Trocar (&n1, &n2);
■ ...
■ }

■ void Trocar (int *pv1, int *pv2)
■ {}
```

São passados os endereços de memória destas variáveis e qualquer alteração feita nessas variáveis no interior da função/procedimento refletirá na variável originalmente passada.



22/07/2007

11

tr.pt/primeira.página



Passagem de Parâmetros - Referência (C++)

```
■ #include <stdio.h>
■ #include <stdlib.h>
■ #include <conio.h>
■ void Trocar (int &v1, int &v2);
■ main()
■ { int n1, n2;
■   printf ("\n Digite o valor de n1 : ");
■   scanf ("%d",&n1);
■   printf ("\n Digite o valor de n2 : ");
■   scanf ("%d",&n2);
```

22/07/2007

12

tr.pt/primeira.página



Passagem de Parâmetros - Referência (C++)

- `printf ("\nAntes de trocar: ");`
- `printf ("\n n1 = %d e n2 = %d ",n1,n2);`
- **Trocar (n1, n2); // Passagem por ...**
- `printf ("\nDepois de trocar: ");`
- `printf ("\n n1 = %d e n2 = %d ",n1,n2);`
- `getch(); }`

22/07/2007

13

[tr.p/primeira.página](#)



Passagem de Parâmetros - Referência (C++)

- **`void Trocar (int &v1, int &v2)`**
- `{ int tmp;`
- `tmp = v1;`
- `v1 = v2;`
- `v2 = tmp;}`

22/07/2007

14

[tr.p/primeira.página](#)



Exercícios Propostos

- 1) Crie um programa que:
- Após definir um vetor **local** de inteiros , chame um procedimento para preencher este vetor com valores aleatórios e chame outro procedimento para percorrer este vetor e mostrar seus elementos.
- 2) Leia um nome (local) com 15 dígitos (no bloco **main**) e crie um procedimento que mostre o nome ao contrário.

22/07/2007

15

[tr.p/primeira.página](#)



Exercícios Propostos

- 3) Em C, o uso de apontadores é essencial por ser o único modo de se conseguir fazer passagem por referência. Como toda passagem de parâmetros em C é feita por valor, o único modo de passagem por referência é passar como parâmetro um apontador para a variável que se deseja passar por referência. Considere por exemplo a função abaixo:
- `void incrementa(int *x) {`
- `++(*x); }`
- Qual o resultado da chamada "incrementa(&y)" onde y é uma variável inteira ?

22/07/2007

16

[tr.p/primeira.página](#)

