



Universidade Luterana do Brasil
ULBRA – Campus Cachoeira do Sul
Pró-Reitoria de Graduação

Tipo de atividade:

Prova (x) Trabalho () ()

Avaliação: G1 () G2 (x)

Substituição de Grau: G1 () G2 ()

Curso: Sistemas de Informação	Disciplina: Algoritmos e Programação II	Data: 10/12/2014
Turma:	Professor(a): Daniela Scherer dos Santos	Peso da Avaliação: 7,0
Acadêmico(a):	n°:	Nota:

Instruções:

- Prova individual e sem consulta;
- Interprete devidamente as questões, visto ser esta uma das habilidades exigidas na avaliação;
- Não utilize mídias eletrônicas, salve seus programas no computador;
- Coloque sua **IDENTIFICAÇÃO** como comentário no início de cada programa;
- Para a resolução da **PARTE I** da prova não é permitido o uso de computador;
- É proibido o uso de quaisquer dispositivos eletrônicos (telefones celulares, smartphones, tablets, notebook ou similares) bem como o uso de pendrives;**
- Para entregar, **CHAME** a professora.

PARTE I

- (1,0 ponto) (adaptado de TRF 2010 - FCC) Observe o fragmento de código abaixo (Algoritmo 1):

```
1. ...
2. int n, x, conta, contB, exp, resp;
3. n = 4;
4. x = 2;
5. resp = x;
6. for(conta=2; conta<=n; conta++)
7. {
8.     exp = x;
9.     for(contB=1; contB<=conta-1; contB++)
10.    {
11.        exp = exp * x;
12.    }
13.    resp = resp + exp;
14. }
15. Console.WriteLine("{0}", resp);
16. ...
```

Algoritmo 1

Qual será o valor da variável *resp* quando a linha 15 for executada?

RESPOSTA: 30

- (0,5 pontos) Assuma o seguinte programa (Algoritmo 2) e considere os endereços de memória na Tabela 1:

```
1. static void Main(string[] args)
2. {
3.     unsafe
4.     {
5.         int x, y;
6.         int* p;
7.         x = 15;
8.         y = 0;
9.         p = &y;
10.        x = *p;
11.    }
12. }
```

Algoritmo 2

Tabela 1

Variável	Endereço de Memória
y	19813
p	2F341
x	18FF3

Complete a Tabela 2 com os resultados de cada expressão após a execução da linha 10 do Algoritmo 2.

Tabela 2

Expressão	x	y	*p	&x	&y	p	&p
Resultado	0	0	0	18ff3	19813	19813	2f341

3. (1,0 ponto) (adaptado de ENADE 2008) Com relação ao algoritmo abaixo (Algoritmo 3), que manipula um vetor de inteiros cujo conteúdo inicial é composto pelos valores: 2, 0, 4, 3 e 1, julgue os itens a seguir:

```

1. static void Main(string[] args)
2. {
3.     int[] v = {2,0,4,3,1};
4.     int I, J, A;
5.     for (I = 0; I <= 3; I++)
6.     {
7.         for (J = 0; J <= 3 - I; J++)
8.         {
9.             if (v[J] > v[J + 1])
10.            {
11.                A = v[J];
12.                v[J] = v[J + 1];
13.                v[J + 1] = A;
14.                Console.WriteLine("{0},{1},{2},{3},{4}", v[0], v[1], v[2], v[3], v[4]);
15.            }
16.        }
17.    }
18. }

```

Algoritmo 3

- Quando as variáveis I e J valerem, respectivamente, 0 e 1, a linha 14 apresentará a sequência de valores 0,2,4,3,1.
- Quando as variáveis I e J valerem, respectivamente, 1 e 0, a linha 14 apresentará a sequência de valores 0,2,3,1,4.
- Quando as variáveis I e J valerem, respectivamente, 1 e 2, a linha 14 apresentará a sequência de valores 0,2,1,3,4.

Assinale a opção correta.

- Apenas um item está certo.
- Apenas os itens I e II estão certos.
- Apenas os itens I e III estão certos.
- Apenas os itens II e III estão certos.
- Todos os itens estão corretos.

4. (1,0 ponto) Observe o fragmento de código abaixo (Algoritmo 4) e apresente quais serão os valores armazenados em cada posição/índice do vetor *r* (iniciando pela posição zero) após a execução do algoritmo.

RESPOSTA: Vetor r = {10,33,20,44,30,55,40,66}

```

1. ...
2. const int N = 4;
3. int[] v1 = {10,20,30,40};
4. int[] v2 = {33,44,55,66};
5. int[] r = new int[N*2];
6. int i;
7. for (i=0; i<N; i++)
8. {
9.     r[2*i] = v1[i];
10.    r[2*i+1] = v2[i];
11. }
12. ...

```


Algoritmo 4

5. (0,5 pontos) Dado o programa em C# a seguir (Algoritmo 5), qual será o valor impresso para as variáveis *v1* e *v2* após a execução da linha 10?

RESPOSTA: V1 = 12 E V2 = 16

```
1. static void Main(string[] args)
2. {
3.     unsafe
4.     {
5.         int v1, v2;
6.         v1 = 2;
7.         v2 = 6;
8.         b(&v1, &v2);
9.         a(v1, v2);
10.        Console.Write("{0} {1}", v1, v2);
11.    }
12. }
13. static void a(int x, int y)
14. {
15.     x = x + 5;
16.     y = y + 2;
17. }
18. static unsafe void b(int* x, int* y)
19. {
20.     *x = *x + 10;
21.     *y = *y + 10;
22. }
```

Algoritmo 5

 Universidade Luterana do Brasil ULBRA – Campus Cachoeira do Sul Pró-Reitoria de Graduação		Tipo de atividade: Prova (x) Trabalho () () Avaliação: G1 () G2 (x) Substituição de Grau: G1 () G2 ()
Curso: Sistemas de Informação	Disciplina: Algoritmos e Programação II	Data: 10/12/2014
Turma:	Professor(a): Daniela Scherer dos Santos	Peso da Avaliação: 7,0
Acadêmico(a):	n°:	Nota:

PARTE II:

O problema descrito abaixo deve ser solucionado utilizando a linguagem de programação C#. Critérios como: organização e clareza do código, correto funcionamento do programa e lógica utilizada para solucionar o problema também serão utilizados na avaliação.

1. (2,0 pontos) A empresa TI Now precisa de um sistema para gerenciar os livros de uma livraria com as seguintes características:
 - a) ao ser executado deverá apresentar um Menu ao usuário informando as opções de operações disponíveis no sistema:
MENU
(1) Cadastrar livros
(2) Relatório de livros
(3) Pesquisar livro por autor
(4) Sair do sistema
 - b) quando a opção 1 for escolhida, uma função deverá ser chamada para cadastrar os dados (título, autor, estilo, código e preço) de todos os N livros da livraria. N poderá ser lido ou definido pelo programador como uma constante;
 - c) quando a opção 2 for escolhida, uma função deverá ser chamada para emitir um relatório com as seguintes informações:
 - Lista de livros cadastrados (título, autor, estilo, código e preço);
 - Total em R\$ em estoque (soma dos preços de todos os livros cadastrados);
 - Os dados do livro de maior preço (o qual deverá ser computado por outra função);
 - d) quando a opção 3 for escolhida, uma função deverá apresentar todas as informações de todos os livros do autor solicitado pelo usuário;
 - e) quando a opção 4 for escolhida, o sistema deverá ser encerrado;
 - f) o sistema deverá usar registro para agrupar os dados dos livros e vetor para armazenar todos os livros cadastrados.

//solução adaptada do aluno Leonardo

```

class Program
{
    struct dados
    {
        public string titulo;
        public string autor;
        public string estilo;
        public int codigo;
        public decimal preco;
    }
    unsafe static void Main(string[] args)
    {
        int x, n;
        bool zero = false;

        Console.WriteLine("Número de livros que deseja cadastrar:");
        n = Convert.ToInt32(Console.ReadLine());
        dados[] vetor = new dados[n];

        do

```

```

{
    x = menu();

    if (x == 1)
    {
        if(zero==false)
        {
            op1(vetor, n);
            zero = true;
        }
        else
            Console.WriteLine("Dados já foram cadastrados!");
    }
    else if (x == 2)
    {
        if (zero == true)
        {
            op2(vetor, n);
        }
        else
        {
            Console.WriteLine("Banco de dados vazio!");
        }
    }
    else if (x == 3)
    {
        if (zero == true)
        {
            op3(vetor, n);
        }
        else
        {
            Console.WriteLine("Banco de dados vazio!");
        }
    }
}
while (x != 4);

}

static int menu()
{
    int x;
    do
    {
        Console.WriteLine("Menu: \n (1) Cadastrar livros \n (2) Relatório de li
vros \n (3) Pesquisar livro por autor \n (4) Sair do sistema \n Opção: ");
        x = Convert.ToInt32(Console.ReadLine());
        if (x<1 || x>4)
            Console.WriteLine("Opção Inválida! Tente novamente!");
    }
    while (x != 1 && x != 2 && x != 3 && x != 4);

    return x;
}

static void op1(dados[] vetor, int n)
{
    int y;

    Console.WriteLine("Cadastro de livros:");
    for (y = 0; y < n; y++)
    {
        Console.WriteLine("Título: ");
    }
}

```

```

        vetor[y].titulo = Convert.ToString(Console.ReadLine());
        Console.WriteLine("Autor: ");
        vetor[y].autor = Convert.ToString(Console.ReadLine());
        Console.WriteLine("Estilo: ");
        vetor[y].estilo = Convert.ToString(Console.ReadLine());
        Console.WriteLine("Código: ");
        vetor[y].codigo = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Preço: ");
        vetor[y].preco = Convert.ToDecimal(Console.ReadLine());
    }
}

static void op2(dados[] vet, int n)
{
    int x;
    decimal y = 0;
    for (x = 0; x < n; x++)
    {
        Console.WriteLine("Título: " + vet[x].titulo);
        Console.WriteLine("Autor: " + vet[x].autor);
        Console.WriteLine("Estilo: " + vet[x].estilo);
        Console.WriteLine("Código: " + vet[x].codigo);
        Console.WriteLine("Preço: " + vet[x].preco);

        y = y + vet[x].preco;
    }
    Console.WriteLine("Preço total: " + y);
    verificaMaiorPreco (vet);
}

static void op3(dados[] vetor, int n)
{
    string aut;
    int z = 0;
    bool achou = false;
    Console.WriteLine("Digite o nome do Autor");
    aut = Console.ReadLine();
    for (z = 0; z < n; z++)
    {
        if (aut == vetor[z].autor)
        {
            Console.WriteLine("Título: " + vetor[z].titulo);
            Console.WriteLine("Autor: " + vetor[z].autor);
            Console.WriteLine("Estilo: " + vetor[z].estilo);
            Console.WriteLine("Código: " + vetor[z].codigo);
            Console.WriteLine("Preço: " + vetor[z].preco);
            achou = true;
        }
    }
    if (achou == false)
        Console.WriteLine ("Autor não encontrado!");
}

static void verificaMaiorPreco(dados[] vet)
{
    decimal m = vet [0].preco;
    int pos=0;
    for (int x = 0; x < vet.Length; x++)
    {
        if (vet[x].preco > m)
        {
            m = vet[x].preco;
            pos = x;
        }
    }
}

```

```
}
Console.WriteLine("Maior preço: ");
Console.WriteLine("Título: " + vet[pos].titulo);
Console.WriteLine("Autor: " + vet[pos].autor);
Console.WriteLine("Estilo: " + vet[pos].estilo);
Console.WriteLine("Código: " + vet[pos].codigo);
Console.WriteLine("Preço: " + vet[pos].preco);
}
}
```