

Universidade Paulista
Curso Superior de Tecnologia

LISTA 2 DE EXERCÍCIOS DE PROGRAMAÇÃO

Nome: Daniel Pessoto Medeiros Guimarães RA: N4286J1

ADS 3º Semestre

Campus: Pinheiros

Exercícios de programação

Professor: Angel

São Paulo

Maio/2020

Questão 1)

1. (.5 Ponto) O trecho de código a seguir foi escrito em C#. Assinale a alternativa que faz a afirmação correta em relação ao código exibido a seguir.

```
1 public class Aluno{
2     public string NomeAluno { get; set; }
3     public Aluno(){
4         this.NomeAluno = "";
5     }
6     public Aluno(string Nome){
7         NomeAluno = Nome;
8     }
9 }
```

- A** () Não é permitido o uso de construtor sem parâmetros junto com o com parâmetros.
- B** () Está faltando o uso de this no segundo método.
- C** () o Atributo NomeAluno viola a recomendação do encapsulamento pois é publico.
- D** () A classe Aluno não compilará.
- E** () Há dois construtores.

Solução:

Alternativa-Correta: E

Justificativa: A classe dispõe de 2 construtores, um sem argumentos e o outro com.

Questão 2)

2. (.5 Ponto) Com relação a linguagem C#, **NÃO** é correto afirmar o que se diz na alternativa ?

- A** () Uma classe não pode ser derivada de mais de uma interface simultaneamente.
- B** () Uma classe não pode ser derivada de mais de uma classe simultaneamente.
- C** () É permitido uma classe chamar outra classe em uma dll.
- D** () Pode-se ter várias classes em uma dll.
- E** () É possível uma classe chamar outra classe em um namespace diferente.

Solução:

Alternativa-Correta: B

Justificativa: C# possui herança simples, ou seja, só pode ter uma classe mãe/pai, diferente de outras linguagens que possuem herança multiplica, que podem herdar de 2 ou mais classes.

Questão 3)

3. (.5 Ponto) O trecho de código a seguir escrito em C# atende de forma muito simplista, uma das recomendações do paradigma da programação orientada a objetos. Assinale a alternativa que faz a afirmação correta em relação ao código abaixo.

```
1 public class Casa{
2     private string Cor;
3     public void SetCor(string nomeCor){
4         this.Cor = nomeCor;}
5     public string GetCor(){
6         return this.Cor;}
7 }
```

- A** ☐ A recomendação atendida de forma simplista é a da reutilização.
- B** ☐ O código não funcionará pois falta o construtor.
- C** ☐ O this não pode ser usado dentro de uma classe.
- D** ☐ A recomendação atendida de forma simplista é a do encapsulamento.
- E** ☐ A recomendação atendida de forma simplista é a da herança.

Solução:

Alternativa-Correta: D

Justificativa: Foi utilizada o encapsulamento no atributo, para que não se tenha acesso direto a este atributo, sendo necessário a utilização dos métodos Get e Set.

Os trechos 1,2 e 3 de código escritos em C# a seguir deverão ser analisados para as questões 4 e 5.

Trecho 1

```
1 namespace Unip.P00I.A2019{
2     public class CalCientifica{
3         public double Cosseno(double graus){
4             double ret = double.MinValue;
5             ret = Math.Cos(graus);
6             return ret;}
7     }
8 }
```

Trecho 2

```
1 namespace Unip.P00I.Turma2019
2 {
3     public class CalAritmetica{
4         public int Subtracao(int n1, int n2){
5             int ret = int.MinValue;
6             ret = n1-n2;
7             return ret;}
8     }
9 }
```

Trecho 3

```
1 using Unip.P00I.Turma2019;
2 namespace MEU_ESPACO_DE_NOMES
3 {
4     class Program{
5         static void Main(string[] args){
6             double graus = 30.0;
7             Unip.P00I.A2019.CalCientifica calc1 = new Unip.P00I.A2019.CalCientifica();
8             Console.WriteLine("Cosseno de {0} e {1}", graus, calc1.Cosseno(30.0));
9             CalAritmetica calc2 = new CalAritmetica();
10            Console.WriteLine("A subtracao de 5 - 3 e = {0}", calc2.Subtracao(5,3));}
11     }
12 }
```

Questão 4)

4. (.5 Ponto) Assinale apenas a alternativa para a qual é possível realizar uma afirmação correta.

- ☒ A () A classe CalAritmetica tem um método.
- ☐ B () A classe calc1 tem dois métodos.
- ☐ C () A classe calc2 tem um método.
- ☐ D () O namespace não pode ser escrito usando o caractere '.' (ponto).
- ☐ E () A classe CalCientifica não pode ser usada em Main pois o namespace não foi declarado.

Solução:

Alternativa-Correta: A

Justificativa: A classe CalAritmetica possui apenas um método, que é o Subtracao(`int` n1, `int` n2).

Questão 5)

5. (.5 Ponto) Assinale apenas a alternativa para a qual é possível realizar uma afirmação correta.

- ☒ A () O objeto calc1 tem dois métodos.
- ☐ B () O objeto calc2 tem um método.
- ☐ C () A classe calc1 tem um método.
- ☐ D () O namespace Unip.POOLA2018 só pode ser usado através da declaração using.
- ☐ E () A classe CalAritmetica é declarada no namespace Unip.POOLA2018.

Solução:

Alternativa-Correta: B

Justificativa: O objeto calc2, possui apenas um método, que é o método da classe CalAritmetica, que seria o método Subtracao(`int` n1, `int` n2).

Questão 6)

6. (.5 Ponto) Sabendo-se que o código a seguir foi escrito em C#.

```
1 public class Calculadora
2 {
3     public static int operando1;
4     public static int operando2;
5
6     public static int Soma()
7     {
8         return operando1 + operando2;
9     }
10
11     public int Soma(int op1, int op2)
12     {
13         Calculadora.operando1 = op1;
14         Calculadora.operando2 = op2;
15         return Calculadora.Soma();
16     }
17 }
```

- I. O método Soma() é um método de instância.
- II. O método Soma(int op1, int op2) é um método de instância.
- III. O atributo operando1 é um atributo de classe.

É correto afirmar-se que?

- ☒ A () Apenas a afirmação II é correta.
- ☐ B () Apenas a afirmação I é correta.
- ☐ C () Apenas a afirmação III é correta.
- ☐ D () As afirmações II e III são corretas.
- ☐ E () As afirmações I e II são corretas.

Solução:

Alternativa-Correta: D

Justificativa: O método Soma(int op1, int op2) é um método de instância, pois para utilizá-lo, é necessário instanciar um objeto do tipo Calculadora, diferente do método Soma(), que é static, que para acessá-lo, basta apenas indicar a classe Calculadora.Soma(). E tanto o operando1 e operando2, são atributos da classe.

Questão 7)

7. (.5 Ponto) Sabendo-se que o código a seguir foi escrito em C#. A classe de nome **ClasseComErro** foi usada em um programa na criação de um objeto. O objeto criado chamou o método `TesteDivisaoPorZero()` cujo retorno foi atribuído a uma variável inteira. Analisando o código a seguir é correto afirmar o que se diz na alternativa ?

```
1 public class ClasseComErro
2 {
3     public int TesteDivisaoPorZero()
4     {
5         int CodErro = 0;
6         try
7         {
8             int denominador = 2;
9             int Numerador = 6;
10            int x = Numerador / denominador;
11            CodErro = 1;
12        }
13        catch (DivideByZeroException e)
14        {
15            CodErro = -1;
16            string err = e.Message;
17        }
18        catch (Exception ex)
19        {
20            CodErro = -2;
21            string err = ex.Message;
22        }
23        return CodErro;
24    }
25 }
```

- A** (☐) O Valor de CodErro retornado será 1.
- B** (☐) O Valor de CodErro retornado será -2.
- C** (☐) O Valor de CodErro retornado será 3.
- D** (☐) O Valor de CodErro retornado será 0.
- E** (☐) O Valor de CodErro retornado será -1.

Solução:

Alternativa-Correta: A

Justificativa: Como não ocorreu nenhum erro, então o retorno é 1.

Questão 8)

8. (5 Ponto) O código a seguir foi escrito em C#. Assinale a alternativa que faz a afirmação correta em relação ao código exibido a seguir.

```
1 public class Pessoa
2 {
3     protected string _TipoDocumento;
4     public string Nome{get;set;}
5     protected string NumeroDocumento
6     { get; set; }
7     public string TipoDocumento
8     { get { return _TipoDocumento; } }
9 }
10
11 public class PessoaJuridica : Pessoa
12 {
13     ValidaCNPJ valCnpj = new ValidaCNPJ();
14     public string CNPJ{
15         set{this.NumeroDocumento = value;}
16         get{return this.NumeroDocumento;}
17     }
18     public PessoaJuridica()
19     {_TipoDocumento = "CNPJ";}
20     public bool CnpjValido()
21     {return
22         valCnpj.IsCnpj(this.NumeroDocumento);
23     }
24 }
```

- I. A classe Pessoa deriva da classe PessoaJuridica.
 - II. A classe PessoaJuridica herda da classe Pessoa.
 - III. A propriedade NumeroDocumento é protegida e não pode ser usada na classe derivada.
- A** () As afirmações II e III são corretas.
- B** () Apenas a afirmação I é correta.
- C** () apenas a afirmação III é correta.
- D** () As afirmações I e II são corretas.
- E** () Apenas a afirmação II é correta.

Solução:

Alternativa-Correta: E

Justificativa: A classe PessoaJuridica herda a classe Pessoa, portanto todos os atributos da classe Pessoa pode ser acessadas pela classe PessoaJuridica.

Questão 9)

9. (.5 Ponto) O trecho de código a seguir que foi escrito em C#, presumindo que todos estejam no mesmo namespace, é correto afirmar o que se diz na alternativa ?

```
1 interface Interface01{
2 int metodo01(int num1);
3 double metodo01(string parametro1);
4 }
5 interface Interface02{
6 string metodo02(char UmaLetra);
7 }
8 class Classe01 : Interface01,
9 Interface02
10 {
11 public int metodo01(int num1)
12 {
13 return num1 + 1;
14 }
15
16 public double
17 metodo01(string parametro1)
18 {
19 double Resultado=9.3;
20 return Resultado;
21 }
22
23 public string
24 metodo02(char UmaLetra)
25 {
26 string Resultado = string.Empty;
27 Resultado="A letra fornecida foi [" +
28 UmaLetra.ToString() + "];
29 return Resultado;
30 }
31 }
```

- I. A classe Classe01 deriva simultaneamente de duas interfaces.
 - II. A classe Classe01 implementa todos os métodos das interfaces das quais deriva.
 - III. A classe Classe01 não está derivando de nenhuma classe.
- ☒ A () Apenas a afirmação III é correta.
- ☐ B () Apenas a afirmação I é correta.
- ☐ C () As afirmações I, II e III são corretas.
- ☐ D () Apenas a afirmação II é correta.
- ☐ E () As afirmações I e II não são corretas.

Solução:

Alternativa-Correta: C.

Justificativa: A classe Classe01 deriva das interfaces Interface01 e Interface02, e por derivar de interfaces tem que implementar os métodos contidos nelas. E não herda de nenhuma outra classe.

Questão 10)

10. (.5 Ponto) Analisado o código do programa a seguir determine a alternativa correta?

```
1 static void Main(string[] args)
2 {
3     int n1 = 3;
4     int n2 = 4;
5     int n3 = 5;
6
7     if (n1 == n2)
8     {
9         Console.WriteLine
10      (" n1 e diferente de n2");
11     }
12     else
13     {
14         Console.WriteLine
15      (" n1 e igual a n2");
16     }
17
18     if (n3 < n2)
19     {
20         Console.WriteLine
21      (" n3 e menor n2");
22     }
23     else
24     {
25         Console.WriteLine
26      (" n3 nao e menor n2");
27     }
28 }
```

- I. O programa exibe a frase " n1 e diferente de n2".
- II. O programa exibe a frase " n1 e igual a n2".
- III. O programa exibe a frase " n3 nao e menor n2".
- ☒ A () As afirmações I e II estão corretas.
- ☒ B () As afirmações II e III estão corretas.
- ☐ C () As afirmações I e II estão erradas.
- ☐ D () As afirmações I, II e III estão erradas.
- ☐ E () As afirmações I e III estão corretas.

Solução:

Alternativa-Correta: B

Justificativa: O que é impresso na tela, são os conteúdos contidos nos else que existe no programa, que são respectivamente: `Console.WriteLine(" n1 e igual a n2");` e `Console.WriteLine(" n3 nao e menor n2");` .

Questão 11)

11. (5 Ponto) Sobre o programa a seguir qual das alternativas é a correta?

```
1 static void Main(string[] args)
2 {
3     string y = "";
4     char ol = 'c';
5     switch (ol)
6     {
7         case '1':
8             y = "ACHEI A OPCAO 1"; goto case 'A';
9             break;
10        case 'A':
11            y = "ACHEI A OPCAO A";
12            break;
13        default:
14            y = "ACHEI A OPCAO 4";
15            break;
16        }
17    }
18    Console.WriteLine("y={0}", y);
19 }
```

- I. Será exibida a frase "ACHEI A OPCAO 4".
II. Será exibida a frase "ACHEI A OPCAO 1".
III. Será exibida a frase "ACHEI A OPCAO A".
- ☒ A () As afirmações I e III estão corretas.
☐ B () Apenas a afirmação III esta correta.
☒ C () As afirmações I e II estão corretas.
☐ D () Apenas a afirmação II esta correta.
☐ E () Apenas a afirmação I esta correta.

Solução:

Alternativa-Correta: B

Justificativa: Apenas imprimiri "ACHEI A OPCAO A", pois no case 'c', ao final dele existe o goto case 'A', do contrário seria exibido "ACHEI A OPCAO 1".

Questão 12)

12. (5 Ponto) O trecho de código a seguir foi escrito em um programa em C#. Analise a lógica deste trecho e assinale a alternativa correta.

```
1 int z=0;
2 while(z < 10){
3     string v = z.ToString();
4     z++;
5 }
6
7 z = 0;
8 while (z > 3){
9     string v = z.ToString();
10    z++;
11 }
```

- I. O segundo loop while nunca será executado.
II. O segundo loop while será executado três vezes.
III. O primeiro loop while será executado dez vezes.
- ☒ A () As afirmações I e III estão corretas.
☐ B () As afirmações I e II estão corretas.
☒ C () As afirmações I e II estão erradas.
☐ D () As afirmações I, II e III estão erradas.
☐ E () As afirmações II e III estão corretas.

Solução:

Alternativa-Correta: A

Justificativa: O segundo nunca será executado, pois não satisfaz a condição de Z que vale 0, ser maior que 3. O primeiro loop é executado 10 vezes.

Questão 13)

13. (.5 Ponto) Analise o programa em C# a seguir, e assinale a alternativa correta.

```
1 static void Main(string[] args){
2     int[] lista;
3     lista = new int[] { 1, 2, 3, 4, 5, 6};
4     int k14 = -2;
5     foreach (int num in lista){
6         k14 += num;
7         if (k14 % 2 == 0) break;
8     }
9     Console.WriteLine("K14={0}",--k14);
10 }
```

- ☐ A () O programa exibirá K14=-2.
- ☐ B () O programa exibirá K14=3.
- ☐ C () O programa exibirá K14=-k14.
- ☐ D () O programa exibirá K14=4.
- ☐ E () O programa exibirá K14=2.

Solução:

Alternativa-Correta: B.

Justificativa: Antes dele entrar no foreach, K14 está valendo -2. Quando entrar no primeiro laço, ele vale -1 = -2 + 1, depois vale 1 = -1 + 2, depois vale 4 = 1 + 3, e finalmente quando chega neste valor, ele satisfaz a condição do IF que o resto da divisão de k14 que vale 4 dividido por 2, seja zero, portando o foreach é encerrado devido ao break, e é impresso K14=3, pois na hora da impressão é decrementado -1 do valor de k14.

Questão 14)

14. (.5 Ponto) Analise o programa em C# a seguir, e assinale a alternativa correta.

```
1 static void Main(string[] args){
2     int D = 4;
3     string Dia = "";
4     if (D == 1) Dia="Domingo";
5     else if (D == 2) Dia = "Segunda";
6     if (D == 3) Dia = "Terca";
7     else if (D == 4) Dia = "Quarta";
8     if (D == 5) Dia = "Quinta";
9     if (D == 6) Dia = "Sexta";
10    else Dia = "Sabado";
11    Console.WriteLine("Hoje e {0}", Dia);
12 }
```

- ☐ A () O programa exibirá Hoje e Quarta.
- ☐ B () O programa exibirá Hoje e Domingo.
- ☐ C () O programa exibirá Hoje e Sabado.
- ☐ D () O programa exibirá Hoje e Quinta.
- ☐ E () O programa exibirá Hoje e Sexta.

Solução:

Alternativa-Correta: C.

Justificativa: Primeiramente a variável Dia = "Quarta", mas na linha 10 o else faz atribuir Dia = "Sábado", portanto é impresso "Hoje e Sabado".

Questão 15)

15. (.5 Ponto) Com relação ao código a seguir, é correto afirmar o que se diz na alternativa ?

```
1 static void Main(string[] args){
2     int z = 1;
3     string v = "33";
4     for (int i = 0; i > 3; i++){
5         do{
6             v = i.ToString() + z.ToString();
7             z++;
8         } while (z < 2);
9     }
10    Console.WriteLine("v={0}",v);
11 }
```

- ☐ A () O programa exibirá V=2+1.
- ☐ B () O programa exibirá V=4.
- ☒ C () O programa exibirá V=3.
- ☐ D () O programa exibirá V=1+1.
- ☐ E () O programa exibirá V=33.

Solução:

Alternativa-Correta: E.

Justificativa: Como o programa nem chega a entrar no laço, pois a condição estabelecida no for não é satisfeita, portanto é impresso V = 33, pois o valor contido em v = "33".

Questão 16)

16. (.5 Ponto) Com relação ao código a seguir, é correto afirmar o que se diz na alternativa ?

```
1 interface IDBBanco
2 {
3     bool Salvar();
4     bool Atualizar();
5     bool Apagar();
6     bool Consultar();
7 }
```

- ☐ A () Como não foi declarada a visibilidade public da interface ela não será visível fora da classe.
- ☐ B () interface é um termo usado específico da linguagem Java não usado em C#.
- ☐ C () Como não foram declaradas as visibilidades public dos métodos eles não serão implementados.
- ☐ D () Todas as classes que derivarem da interface IDBBanco, devem implementar os métodos Salvar(), Atualizar(), Apagar(), Consultar().
- ☐ E () interface é um termo usado apenas na UML.

Solução:

Alternativa-Correta: D.

Justificativa: Toda classe que derivar da interface IDBBanco, são obrigado a implementar os métodos desta interface.

Questão 17)

17. (2 Pontos) No espaço abaixo crie as classes **Pessoa**, **PessoaFisica**, **PessoaJuridica** e a interface **ValidarDocumento**. As classes **PessoaFisica** e **PessoaJuridica** derivam da classe **Pessoa** e da interface **ValidarDocumento**. A classe **Pessoa** deve ser abstrata e conter uma propriedade tipo string de nome **NomePessoa** de leitura e escrita. A interface **ValidarDocumento** deve ter uma assinatura de nome **Validar** que retorne um booleano. Na classe **Pessoa** crie um método abstrato de nome **CalculeIR** que receba como argumento um inteiro representando o ano da declaração de imposto de renda e que retorne um valor double com o valor do imposto a pagar, a implementação do método deve ser apenas o mínimo necessário para compilar o código, por exemplo um return true ou return 1.0.

Escreva um trecho código na linguagem *C#* que implemente a relação descrita para as entidades mencionadas. Não se preocupe com os métodos Main ou com os namespaces, apenas mostre em *C#* como é o relacionamento das entidades mencionadas com a implementação mínima necessária para o código compile para os métodos mencionados.

Solução:

```
abstract class Pessoa {
    public string NomePessoa { get; set; }

    public abstract double CalculeIR(int ano);
}

class PessoaFisica : Pessoa, ValidarDocumento {
    public override double CalculeIR(int ano) {
        return 1.0;
    }

    public bool Validar() {
        throw new NotImplementedException();
    }
}

class PessoaJuridica : Pessoa, ValidarDocumento {
    public override double CalculeIR(int ano) {
        return 1.0;
    }

    public bool Validar() {
        throw new NotImplementedException();
    }
}

interface ValidarDocumento {
    bool Validar();
}
```