

<b>P.PORTO</b>	ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO E GESTÃO	Tipo de Prova <b>Exame Época Normal</b>	Ano lectivo 2020/2021	Data 28-6-2019
		CURSO <b>Licenciatura em Engenharia Informática e Licenciatura em Segurança Informática em Redes de Computadores</b>	Hora 10:00	
		Unidade Curricular <b>Paradigmas de Programação</b>	Duração 2:00 horas	
Observações O exame é sem consulta				

## Grupo I (8 valores)

Nota: Respostas erradas não descontam.

a) Considerando que: (2 valores)

A e E são **classes** e que B e D são **interfaces**, **quais** das seguintes opções são verdadeiras:

- i) Interface F extends E{ }
- ii) Interface F implements B,D{ }
- iii) Interface F implements D { }
- iv) Interface F extends D{ }
- v) Interface F extends B,D{ }

b) O **encapsulamento** em Java é: (2 valores)

- i) Uma forma de manter um conjunto de classes associadas unicamente a um package, não suportando a sua visibilidade com outros *packages*
- ii) A capacidade de uma classe herdar métodos com o modificador de acesso *protected*;
- iii) Um mecanismo que permite reimplementar o comportamento de um método numa subclasse
- iv) É uma forma de garantir que os valores das variáveis de uma classe não podem ser modificados
- v) Nenhuma das anteriores

c) Em Java, um método declarado como *protected*... (2 valores)

- i) só pode ser acedido dentro da própria classe ou por classes derivadas
- ii) não pode ser redefinido
- iii) não pode chamar outros métodos que também não sejam *protected*
- iv) pode ser acedido por classes da mesma package
- v) só pode ser usado por construtores

d) Em Java, uma classe abstrata: (2 valores)

- i) não pode fornecer a implementação dos métodos que declara;
- ii) pode fornecer a implementação para alguns métodos, mas obrigatoriamente tem de possuir pelo menos um método abstrato.
- iii) não pode ser instanciada
- iv) não pode ser especializada/derivada
- v) nenhuma das anteriores

<b>P.PORTO</b> <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO E GESTÃO</small>	Tipo de Prova <b>Exame Época Normal</b>	Ano lectivo <b>2020/2021</b>	Data <b>28-6-2019</b>
	CURSO <b>Licenciatura em Engenharia Informática e Licenciatura em Segurança Informática em Redes de Computadores</b>	Hora <b>10:00</b>	
	Unidade Curricular <b>Paradigmas de Programação</b>	Duração <b>2:00 horas</b>	

## Grupo II (6 valores)

Considere o seguinte excerto de código:

```

public class Bicycle {
    public int cadence;
    public int gear;
    public int speed;

    public Bicycle(int startCadence, int startSpeed, int startGear) {...}

    (...)
    public void printDescription(){
        System.out.println("\nBike is " + "in gear " + this.gear
        + " with a cadence of " + this.cadence +
        " and travelling at a speed of " + this.speed + ". ");
    }
}

public class MountainBike extends Bicycle {
    private String suspension;

    public MountainBike(...){
        ...
    }

    public String getSuspension(){...}

    public void setSuspension(String suspensionType) {...}

    public void printDescription() {
        super.printDescription();
        System.out.println("The " + "MountainBike has a" +
        getSuspension() + " suspension.");
    }
}

public class TestBikes {
    public static void main(String[] args){
        Bicycle bike01, bike02;

        bike01 = new Bicycle(20, 10, 1);
        bike02 = new MountainBike(20, 10, 5, "Dual");

        bike01.printDescription();
        bike02.printDescription();
    }
}

```

- Qual é o output apresentado pela invocação de cada um dos métodos **printDescription** na classe **TestBikes**? (2 valores)
- Considere que se pretende comparar duas **Bicycle**. Considere que as **Bicycle** possuem um novo atributo: **identifier** (do tipo int), utilizado para comparar se duas **Bicycle** são iguais. Realize as alterações necessárias nas classes apresentadas, de forma que seja possível comparar duas instâncias de **Bicycle**. Na classe **TestBikes**, apresente um exemplo. (4 valores)

## Grupo III (6 valores)

Considere o seguinte domínio:

Ao longo dos anos, a empresa 4Patas tem disponibilizado um conjunto de serviços que privilegiam o tratamento de animais de estimação.

O processo inicia-se com o contacto entre o cliente e o colaborador da empresa. O cliente informa o colaborador, indicando os dados do animal e os serviços que pretende que o animal receba. Atualmente a empresa disponibiliza os seus serviços para cães e gatos, não excluindo a possibilidade de num futuro próximo, abranger outro tipo de animais.

<b>P.PORTO</b> <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO E GESTÃO</small>	Tipo de Prova <b>Exame Época Normal</b>	Ano lectivo <b>2020/2021</b>	Data <b>28-6-2019</b>
	CURSO <b>Licenciatura em Engenharia Informática e Licenciatura em Segurança Informática em Redes de Computadores</b>	Hora <b>10:00</b>	
	Unidade Curricular <b>Paradigmas de Programação</b>	Duração <b>2:00 horas</b>	

Para cada animal, é registado o nome, data de nascimento e se o animal possui a vacinação atualizada. Em cada visita é registado o peso do animal.

Especificamente para os cães, a empresa tem a necessidade de armazenar o seu porte (pequeno, médio ou grande) e para os gatos é necessário identificar o tipo de gato que irá receber: gato de interior ou gato de exterior.

Para a marcação da reserva, o colaborador começa por registar a data e os dados do animal, questionando posteriormente o cliente relativamente aos serviços que pretende incluir da lista que a empresa disponibiliza. Por exemplo: Passeios, Alojamento, Transporte, Banho, Tosquia ou Sessão Fotográfica.

Os serviços podem ser adquiridos individualmente ou, por vezes, em grupos. Por exemplo, o pacote básico inclui: Passeio, Banho e Tosquia.

É ainda necessário controlar o estado da reserva:

- Quando a reserva é inserida, fica ativa;
- A reserva pode ser cancelada;
- A reserva pode ser remarcada;
- A reserva fica concluída após a prestação do serviço.

- Apresente as **interfaces** (pode optar pela utilização de um diagrama de classes) que considera necessárias para representar o problema apresentado (3 valores)
- Apresente uma classe **GestaoReservas** (não tem que implementar as interfaces anteriormente apresentadas) com os seguintes métodos:
  - Adicionar **reservas**, considerando que não podem existir duas reservas iguais armazenadas. O método deverá retornar um valor que sinalize o sucesso/insucesso da operação. (1,5 valores)
  - Disponibilizar uma representação textual das reservas que ainda não foram concluídas. (1,5 valores)