



Universidade Federal do Ceará  
Campus de Quixadá  
Projeto Detalhado de Software  
Paulyne Jucá  
Primeira Avaliação (AP 1)

NOME: \_\_\_\_\_

MODELAGEM (resposta em figura PNG ou JPEG)

- 1) (2 pontos) Sabendo que no seu problema existem bichos com diferentes comportamentos, como você faria a segregação da interface “IAnimais” apresentada abaixo? Sua resposta deve ser um **diagrama de classes** com as interfaces resultantes e seus métodos.

Obs: pode considerar que todos os bichos dormem, comem alguma coisa e emitem som.

```
public interface IAnimais{  
    public void correr();  
    public void nadar();  
    public void voar();  
    public void comerCarne();  
    public void dormir(); ->  
    public void emitirSom(); ->  
    public void amamentar();  
    public void colocarOvos();  
}
```

- 2) (3 pontos) Modele a realização do caso de uso “adicionar um livro” descrito abaixo utilizando um **diagrama de sequência**. O diagrama deve considerar todo o processo indo desde a interface até o serviço de armazenamento de dados.

Caso de uso:

- A bibliotecária deve adicionar um novo livro no sistema. Para isso, ela informa, o título (string), a edição (string), ano (data), autor (selecionando de uma lista pré-cadastrada) e a editora (selecionando de uma lista pré-cadastrada). Depois de informar os dados, ela confirma a adição do novo livro no sistema.
- Considere que a bibliotecária já está logada no sistema e que o livro a ser adicionado tem sempre apenas 1 autor.

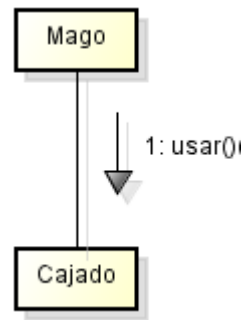
PROGRAMAÇÃO

- 3) (3 pontos) Levando em consideração o diagrama de comunicação e o código abaixo que mostram uma classe Mago acessando diretamente uma classe Cajado, refatore o código das classes utilizando o princípio de inversão de dependência para permitir que diferentes armas (por exemplo, cajado e anel) possam ser usados por esse mesmo Mago. Sua resposta deve ser o **código das classes** (posicionadas corretamente em pacotes levando em consideração que no futuro as diferentes armas também poderão ser usadas por outros personagens) que implementa sua solução. (3 pontos)

```

public class Mago
{
    private Cajado cajado;
    public void Atacar()
    {
        cajado.usar();
    }
}

```



- 4) (2 pontos) Uma criança foi na loja de doces para comprar bombons. A loja vende pacotes pequenos (com 10 unidades), médios (com 25 unidades) e grandes (50 unidades). A criança comprou toda a sua mesada em pacotes de diversos tamanhos e agora deseja saber quantos bombons comprou. Levando em consideração os conceitos de polimorfismo e herança, refatore o código na funcionalidade que calcula a quantidade de bombons para ajudar a criança a contar.

```

import java.util.ArrayList;
public class Compra {
    private ArrayList<SacoDeBombom> sacosDeBombons;

    public Compra() {
        sacosDeBombons = new ArrayList< SacoDeBombom >();
    }

    public int contaQuantidadeBombons () {
        int bombons = 0;
        for (SacoDeBombom sb: sacosDeBombons) {
            if (sb.leTipo().equals("pequeno")){
                bombons += 10;
            } else if (sb.leTipo().equals("medio")){
                bombons += 25;
            } else if (sb.leTipo().equals("grande")){
                bombons += 50;
            }
        }
        return bombons;
    }

    public void adicionaSacoBombom(SacoDeBombom sacosDeBombom){
        this.sacosDeBombons.add(sacosDeBombom);
    }
}

public class SacoDeBombom {

    String tipo;
    String marca;
}

```

```
public SacoDeBombom(String marca, String tipo){  
    this.marca = marca;  
    this.tipo = tipo;  
}  
  
public String leTipo(){  
    return this.tipo;  
}  
}
```