

**Questão 1.** Considere um sistema que simula o preparo de uma xícara de chá. Esse sistema tem as seguintes classes descritas abaixo.

```
/**
 * Essa classe representa os passos envolvidos no
 * preparo de uma xícara de chá.
 */

public class XicaraChá {
    public static final int DEFAULT_INFUSAO = 60;

    private InfusorDeChá infusor;
    private Água água;
    private int tempoInfusao = DEFAULT_INFUSAO;

    public void infundirChá(InfusorChá infusor) {
        this.infusor = infusor;
    }

    public void adicionaÁgua(Água água) {
        this.água = água;
    }

    public void setTempoInfusao(int tempoInfusao) {
        this.tempoInfusao = tempoInfusao;
    }

    public void infusao() {
        int x = 0;

        System.out.print("Chá em infusao por " + tempoInfusao + "
segundos");
        try {
            while (x < tempoInfusao) {
                System.out.print(".");
                x++;
                Thread.sleep(1000);
            }
        }
        catch (InterruptedException interruptedException)
        {
            System.err.println(interruptedException.getMessage());
        }

        System.out.println("Seu chá esta pronto!");
    }
}
```

```
/**
 * Uma classe representando um Cha.
 */

public class Cha {
    private String sabor;

    public Cha(String sabor) {
        this.sabor = sabor;
    }

    public String getsabor() {
        return sabor;
    }
}
```

```
/**
 * Realiza a adicao do cha no infusor.
 */

public class InfusorDeCha {
    private static final int TEMPO_PARA_ADICAO = 5;
    private Cha cha;

    public void adicionaCha(Cha cha) {
        int x = 0;
        this.cha = cha;

        System.out.print("Adicionando " + cha.getsabor() + " ao infusor");
        try {
            while (x < TEMPO_PARA_ADICAO) {
                System.out.print(".");
                Thread.sleep(1000);
                x++;
            }
        } catch (InterruptedException interruptedException)
        {
            System.err.println(interruptedException.getMessage());
        }
    }
}
```

```

/**
 * Representa a agua fervente.
 */

public class Agua {
    private static final int TEMPO_DE_FERVURA = 20;
    int x = 0;

    // simiula a agua fervendo
    public void ferveAgua() {
        System.out.print("Fervendo agua por 20 segundos.");
        try {
            while (x < TEMPO_DE_FERVURA) {
                System.out.print(".");
                x++;
                Thread.sleep(1000);
            }
        }
        catch (InterruptedException ie)
        {
            System.err.println(interrupted.getMessage());
        }
    }
}

```

```

/**
 * Classe cliente.
 */

public class PreparoDeCha {
    public static void main(String[] args) {
        XicaraCha meuCha = new XicaraCha();
        Agua agua = new Agua();
        InfusorDeCha infusor = new InfusorDeCha();
        Cha cha = new Cha("Cha Verde");

        infusor.adicionaCha(cha);

        agua.ferveAgua();

        meuCha.adicionaAgua(agua);

        meuCha.setTempoInfusao(15);

        meuCha.infusao();
    }
}

```

- Crie um projeto Java no Eclipse (ou no NetBeans, se preferir) e teste o código acima.
- Desenhe o diagrama de classes UML para o sistema.
- Observando o diagrama construído no item b), qual é o problema encontrado no projeto e qual é o padrão mais adequado para resolvê-lo?
- (Re)Desenhe o diagrama de classes resultante da aplicação do padrão apontado em c) ao projeto inicial.
- Altere o código dado para que este reflita as mudanças realizadas em d).

**Questão 2.** Resolva os exercícios propostos em sala de aula para os Padrões de Projeto Façade, Singleton e Mediator.