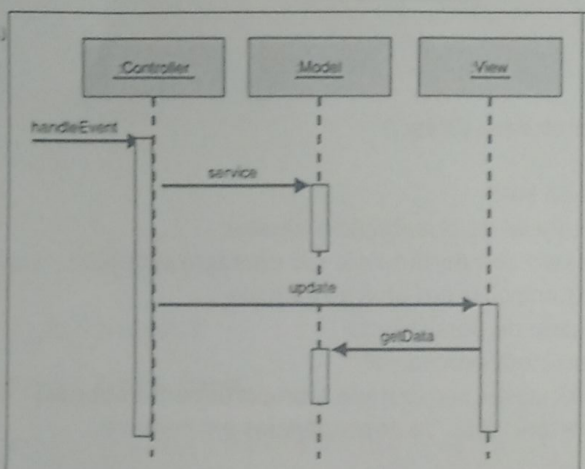
38/45 = 8,4

3ª Avaliação de Engenharia de Software – Engenharia da Computação
Prof. César Olavo

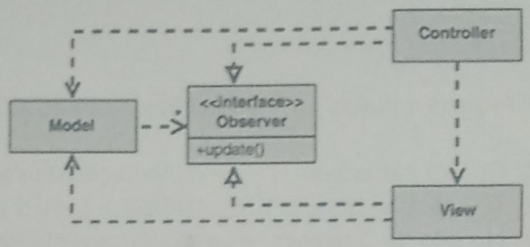
Nome: NICOLAS HOLANDA 29/05/2019

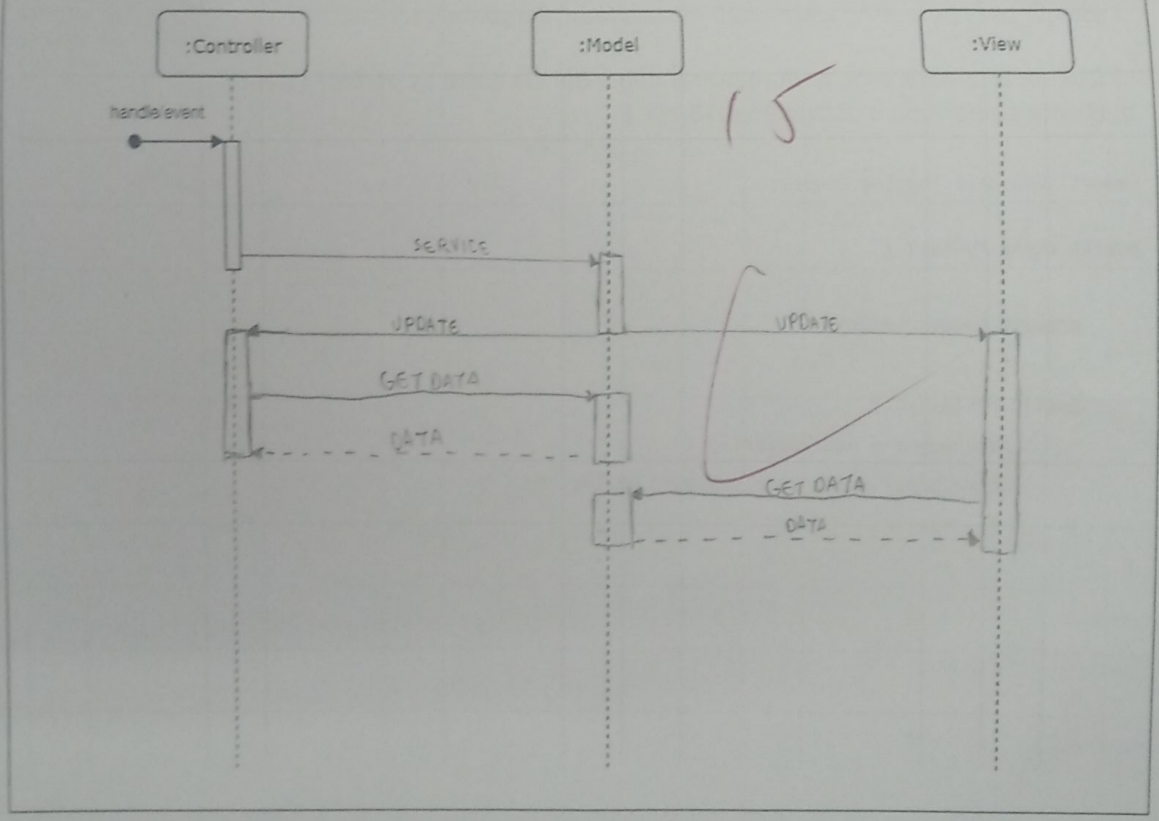
1) Em seu clássico livro *Application Programming in Smalltalk-80: How to use Model-View-Controller (MVC)*, Steve Burbeck descreve duas variações do *MVC*: um modelo *passivo* e um modelo *ativo*. Considerando-se que funcionamento do primeiro tipo pode ser visto na fig. ***a*** abaixo e que o segundo modelo (ativo) incorpora o uso do design pattern *Observer* em relação ao primeiro (passivo), conforme fig. ***b***, explique o funcionamento do modelo ativo. Utilize, para tal, o diagrama de sequência da fig. ***c***. Sugestão: use os métodos dos itens a e b. (15 esc.)

a (PASSIVO)



b (ATIVO)

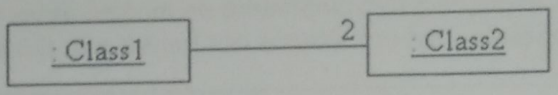




15

2ª Desenhe um diagrama de estado de uma janela de aplicativo desktop que se comporta da maneira esperada (p. ex. se estiver maximizada ou restaurada e for gerado o evento minimizar, ela será minimizada; se estiver minimizada, um clique no ícone da janela fará a mesma retornar ao estado em que se encontrava quando foi minimizada (maximizada ou restaurada), etc. Além de estados e transições, o diagrama deve incluir os pseudo-estados necessários. (10 esc.)

3ª O diagrama abaixo está correto? Em caso negativo, corrija-o. (3 esc.)



4ª Associe colunas em relação aos padrões arquiteturais: (8 esc.)

(1) Quatro camadas	(4) Variação do padrão MVC
(2) Transaction Scripts	(3) Utiliza todo o potencial da orientação a objetos
(3) Modelo de Domínio	4 (8) Utilizam objetos sem comportamento (só guardam estado)
(4) Model-delegate	(1) Combina View e Controller em uma só camada
(5) Injeção de dependência	(1) Implementa camada de persistência
	(2) Promove uma estrutura procedural
	2 (6) É comum possuir classes sem estado (comportamento apenas)
	(5) É alternativa à instanciação de dependências em runtime.

5ª No seu primeiro dia no emprego, seu chefe pede para fazer o teste unitário de MyClass, abaixo. Percebendo o forte acoplamento entre as classes envolvidas, você sugere ao seu chefe modificar o código, para permitir o teste unitário solicitado. Pergunta-se:

a) Que técnica você usou para reduzir a dependência entre as partes? (1 esc)
b) Mostre como ficou o código final. (8 esc.)

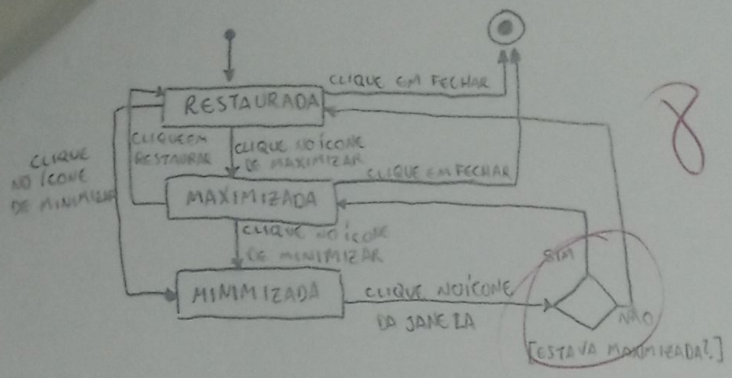
```
import java.util.logging.Logger;

public class MyClass {

     private Logger logger;

     public MyClass() {
          this.logger = new logger();
          logger.info("Isso é uma messagem de log.")
     }
}
```

CANDA



8

(5) a) INJEÇÃO DE DEPENDÊNCIAS

1

b)

```
IMPORT JAVA.UTIL.LOGGING.LOGGER

PUBLIC CLASS MYCLASS {

    PRIVATE LOGGER LOGGER;

    PUBLIC MYCLASS ( LOGGER LOGGER, STRING INFO) {
        THIS.LOGGER = LOGGER;
        LOGGER.INFO (INFO);
    }
}
```

8