

Aufgabe 1

Bedingungs-/Ereignisüberdeckung:

a)

c & (a!b) (4 Testfälle) nur benötigte Ergebnisse in der Tabelle angeben

b)

(a ! b) & c & !d (5 Testfälle) nur benötigte Ergebnisse in der Tabelle angeben

Aufgabe 2:

a)

Anweisungsüberdeckung

Zweigüberdeckung

Pfadüberdeckung k = 1

Tabelle war gegeben, man musste nur noch die Werte für w, x, y und z sowie die Anweisungsfolge angeben.

Für Zweigüberdeckung und Pfadüberdeckung musste man nur noch Fälle angeben, die in den vorherigen Überdeckungen noch nicht angegeben wurden.

b)

Pfadüberdeckung k = 3

Tabelle war gegeben, man musste nur noch die Werte für v, w, x sowie die Anweisungsfolge angeben.

Aufgabe 3:

Klassenadapter

Klasse Connect soll einen Klassenadapter bereitstellen, Klasse Bla und Interface Expected gegeben.

```
public interface Expected {  
    public abstract void add(Object o);    // Fügt das übergebene Objekt in die Datenstruktur  
    ein,  
                                           // falls dieses noch nicht vorhanden ist.  
    public abstract int length();         // Gibt die Anzahl der gespeicherten Elemente  
    zurück.  
    public abstract Object[] getAll();    // Gibt die gespeicherten Objekte in einem  
    Feld zurück.  
}
```

```
public class Bla {  
    public void add(Object o)  
    public Boolean contains(Object o)  
    public Object[] getAll()  
}
```

a)

UML-Klassendiagramm zu der gegebenen Situation angeben.

b)

Implementierung des Klassenadapters Connect angeben.

Aufgabe 4

Dekorierer

Eine Person kann als Mitarbeiter an der Uni angestellt sein.

Ein Mitarbeiter hat einen Vornamen und einen Nachnamen.

Eine Person kann als Dozent Vorlesungen halten. Eine Vorlesung hat ein Attribut ???.

Eine Person kann als Leiter an Projekten beteiligt sein. Ein Projekt hat ein Attribut ???.

a)

UML-Klassendiagramm zu der beschriebenen Situation angeben. Die Datentypen der Attribute sollten nicht angegeben werden.

b)

Skizzieren, wie eine Implementierung aussehen könnte, auch als informeller Text möglich.

Aufgabe 5

Kompositum

Eine Akte besteht aus Verträgen, Notizen, Urkunden und anderen Akten.

Die Dokumente (Verträge, Notizen, Urkunden) haben jeweils einen Verfasser.

a)

UML-Klassendiagramm zu der beschriebenen Situation angeben. Die Datentypen der Attribute sollten nicht angegeben werden.

b)

Skizzieren, wie eine Implementierung aussehen könnte, auch als informeller Text möglich.

Aufgabe 6

Strategie

```
public interface Strategy {  
    int apply(int x, int y);  
}
```

```
public class Application {  
    public void applyStrategy(int[] w, Strategy s){  
        for (int i = 0; i < w.length; i++){  
            w[i] = s.apply(w[i],i);  
        }  
    }  
}
```

a)

Implementierung einer Klasse Int angeben, die das Interface Strategy implementiert und die Funktion $2 \cdot i \cdot w$ berechnet.

b)

Implementierung einer Klasse SetSum angeben, die das Interface Strategy implementiert und $w[i]$ auf die Summe von $w[0]$ bis $w[i]$ setzt.

Aufgabe 7

Observer

Sequenzdiagramm ergänzen. Man durfte davon ausgehen, dass die Observer schon angemeldet waren.

Aufgabe 8

Eine MediBib besteht aus mehreren Medien und Nutzer, die die Medien nutzen.

Ein Nutzer kann sich mehrere Medien ausleihen.

Spezielle Medien sind Videos mit Format-Angabe und OnlineBooks mit ???.

Jedes Medium gehört zu genau einem Themenbereich.

Ein Medium kann nur von einem Nutzer gleichzeitig ausgeliehen werden.

Ein Nutzer hat eine Kennnummer.

Ein Nutzer hat zu jedem Themenbereich eine Merkliste, in der die Medien gespeichert werden, die sich der Nutzer ausleiht.

Klassendiagramm zu der beschriebenen Situation zeichnen. Die Datentypen der Attribute sollten nicht angegeben werden.