

Department of Software Engineering  
RWTH Aachen University  
Prof. Dr. rer. nat. Bernhard Rumpe  
Dipl.-Inform. Achim Lindt  
M.Sc. RWTH Dimitri Plotnikov

*Software Engineering*  
Probeklausur  
WS 2013/14  
22.01.2014

## Probeklausur Softwaretechnik

### Hinweise

Jedes Blatt ist in der Kopfzeile mit Ihrem Namen und Ihrer Matrikelnummer zu versehen. Die Aufgaben sind in den dafür vorgesehenen Freiräumen oder auf den freien Seiten der Blätter zu lösen. Sollte der Platz nicht ausreichen, so können Zusatzblätter verwendet werden. Diese sind dann ebenfalls mit Namen und Matrikelnummer zu kennzeichnen.

Außer (dokumentenechten) Schreibgeräten (kein Bleistift o.ä.; außerdem kein Rot oder Grün) sind keine weiteren Hilfsmittel erlaubt.

Die Klausur besteht aus insgesamt 6 Aufgaben in Deutsch auf 16 Seiten. Die Bearbeitungszeit beträgt **90 Minuten**.

**Hinweis: Die Bearbeitungszeit der tatsächlichen Klausur beträgt 120 Minuten und enthält zusätzliche Teilaufgaben!**

Viel Erfolg!

### Persönliche Angaben

Name, Vorname

.....

Matrikelnummer

.....

Studiengang

.....

Unterschrift

.....

### Punkte

	Aufgabe 1	Aufgabe 2	Aufgabe 3	Aufgabe 4	Aufgabe 5	Aufgabe 6	Summe
	15	20	25	10	15	15	100
Punkte							
Zeichen							

Name:

Matrikelnummer:

## Aufgabe 1 – UML in der Analyse (15 Punkte)

Lesen Sie die *komplette* Aufgabe *gründlich*, bevor Sie mit der Bearbeitung beginnen!

Sie befinden sich gegenwärtig in der Anforderungsanalyse des Computerspiels *Mega Maria Sisters*, in dem ein bzw. zwei Spielfiguren von ein bzw. zwei Spielern durch ein Spielgebiet zu einem Ziel geführt werden müssen. Dabei müssen Gegner und Hindernisse überwunden.

### Teilaufgabe a)

Verschaffen Sie sich einen ersten strukturierten Überblick über den Funktionsumfang des Spiels, indem Sie ein geeignetes Use-Case-Diagramm auf Grundlage der folgenden textuellen Beschreibung erstellen.

*Die zentrale Funktion von Mega Maria Sisters ist das Spielen. Es können entweder ein Spieler (Spieler A) im Einzelspielermodus oder zwei Spieler (Spieler A und Spieler B) im Zweispielermodus spielen. Vor dem Spielen muss Spieler A ein Spielgebiet auswählen. Nach dem Spielen im Einzelspielermodus darf sich Spieler A in der Highscore-Tabelle eintragen, falls er das Spielgebiet erfolgreich beenden konnte.*

Name:

Matrikelnummer:

Name:

Matrikelnummer:

## Aufgabe 2 – UML im Entwurf (20 Punkte)

Lesen Sie die *komplette* Aufgabe *gründlich*, bevor Sie mit der Bearbeitung beginnen!

Nach einer gründlichen Analyse, befinden Sie sich nun im Entwurf von *Mega Maria Sisters*, in dem Sie sich genauere Gedanken über die Struktur und Dynamik des Spiels machen wollen.

### Teilaufgabe a)

Formalisieren Sie die folgende textuelle Beschreibung eines Spielgebietes, indem Sie ein geeignetes Klassendiagramm erstellen. Benutzen Sie dabei mindestens einen Rollennamen und mindestens einen Assoziationsnamen. (Sie dürfen Setter und Getter ignorieren.)

*Ein Spielgebiet hat einen Namen. Ein Spielgebiet besteht aus mindestens einem Spielelement und ein oder zwei Spielfiguren. Ein Spielelement hat einen x-Wert und einen y-Wert vom Typ long, die seine Position im Spielgebiet bestimmen. Spielelemente sind entweder Formen, Gegenstände oder Gegner. Formen können dabei mit beliebig vielen anderen Formen verbunden sein. Genau ein Gegner ist dem Spielgebiet als der sogenannte Endgegner bekannt. Spielfiguren können Gegenstände erhalten. Gegner können getötet werden. Gegenstände können genutzt werden.*

### Teilaufgabe b)

Formalisieren Sie das folgende Szenario, indem Sie ein geeignetes Sequenzdiagramm erstellen, das sich soweit möglich an Ihrem Klassendiagramm aus Teilaufgabe a) orientiert.

*Es existieren ein Spielgebiet  $s$  mit einer Spielfigur  $f$  und einem Gegner  $g$ . Das Spielgebiet instanziert einen Gegenstand namens `bombe` und übergibt es der Spielfigur. Danach benutzt die Spielfigur den Gegenstand. Die Bombe tötet den Gegner, wodurch beide Objekte zerstört werden.*

Name:

Matrikelnummer:

Name:

Matrikelnummer:

### Aufgabe 3 – UML-Zustandsdiagramme (25 Punkte)

Lesen Sie die *komplette* Aufgabe *gründlich*, bevor Sie mit der Bearbeitung beginnen!

In *Mega Maria Sisters* befindet sich eine Spielfigur immer in einem bestimmten Zustand. Formalisieren Sie den Zustandsraum einer Spielfigur, sowie alle Zustandsübergänge, indem Sie zu der folgenden textuellen Beschreibung ein geeignetes Zustandsdiagramm erstellen. Benutzen Sie dafür ausschließlich die folgenden Ereignisse, Bedingungen und Aktionen.

#### Ereignisse

- Spielfigur wird getroffen (`Hit`)
- Spielfigur fällt in einen Abgrund (`Fall`)
- Spielfigur benutzt ein Bonbon (`Bonbon`)
- Spielfigur benutzt „Power Mond“ (`PowerMoon`)
- Zeit von `hh:mm:ss` ist verstrichen (`after hh:mm:ss`)

#### Bedingungen

- Die Anzahl der Leben des Spielers ist größer als Null (`Lives > 0`)
- Die Anzahl der Leben des Spielers ist gleich Null (`Lives == 0`)

#### Aktionen

- Spielfigur wächst auf doppelte Größe (`Grow`)
- Spielfigur schrumpft auf normale Größe (`Shrink`)
- Spielfigur stirbt (`Die`)
- Aktuelles Spiel wird mit normal großer Figur neu gestartet (`Restart`)
- Das Spiel ist „Game Over“ (`GameOver`)

*Zu Beginn eines Spiels ist die Spielfigur normal groß. Wenn die Spielfigur normal groß ist und den Gegenstand „Bonbon“ benutzt, wächst sie zu doppelter Größe. Wenn die Spielfigur von einem Gegner getroffen wird, schrumpft sie zu normaler Größe, falls sie doppelte Größe hatte, oder stirbt, falls sie normal groß war.*

*Wenn die Spielfigur den Gegenstand „Power Mond“ benutzt, wächst sie zu doppelter Größe, falls sie vorher nur normale Größe hatte, und wird für 10 Sekunden unbesiegbar. Während sie unbesiegbar ist, kann sie nicht von Gegnern getroffen werden.*

*Immer wenn die Spielfigur in einen Abgrund fällt, stirbt sie. Nachdem eine Spielfigur gestorben ist, wird das aktuelle Spiel neu gestartet, falls der Spieler noch Leben hat. Hat der Spieler allerdings keine Leben mehr, ist das Spiel „Game Over“ und danach kein Zustand mehr aktiv.*

Name:

Matrikelnummer:

Name:

Matrikelnummer:

## Aufgabe 4 – Entwurfsmuster (10 Punkte)

Ein Spielgebiet in *Mega Maria Sisters* enthält Formen. Gehen Sie nun davon aus, dass sie bei Formen zwischen einfachen Formen und komplexen Formen unterscheiden wollen. Eine komplexe Form ist dabei eine Form, die sich aus anderen Formen zusammensetzt. Eine einfache Form ist hingegen atomar.

Formen sollen dabei die Methode `getDimension() : Dimension` implementieren, deren Aufrufer dabei nicht wissen sollen muss, ob er sie auf einer einfachen oder komplexen Form aufruft.

Welches Entwurfsmuster der „Gang of Four“ ist zur Lösung dieser Anforderung geeignet?

- a. Nennen Sie den Namen des Musters.
- b. Zeichnen Sie das Klassendiagramm des Musters (basierend auf dem gegebenen Beispiel) und kennzeichnen Sie seine Rollen.



Name:

Matrikelnummer:

Name:

Matrikelnummer:

## Aufgabe 5 – Softwaretests (15 Punkte)

Gegeben sei folgender Java-Code. Die Methode `sort(int[] a)` nimmt ein Array von Integers entgegen sortiert es mithilfe eines Selection-Sort-Algorithmus.

```
public static int[] sort(int[] a) {  
    int i = 0; // 1  
    while (i < a.length - 1) { // 2  
        int m = i; int j = i+1; // 3  
        while (j < a.length) { // 4  
            if (a[j] < a[m]) { // 5  
                m = j; // 6  
            }  
            int t = a[i]; a[i] = a[m]; a[m] = t; j++; // 7  
        }  
        i++; // 8  
    }  
    return a; // 9  
}
```

- Erstellen Sie den Kontrollflussgraphen für diese Methode.
- Geben Sie anschließend ein minimales Eingabearray für einen Anweisungsüberdeckungstest an. Geben Sie für diese Angabe die Sequenz der besuchten Knoten aus ihrem Kontrollflussgraphen an.
- Geben Sie ein minimales Eingabearray für einen vollständigen Pfadüberdeckungstest an, falls Sie eines angeben können, oder geben Sie eine Begründung an, falls sie keines angeben können.

Name:

Matrikelnummer:

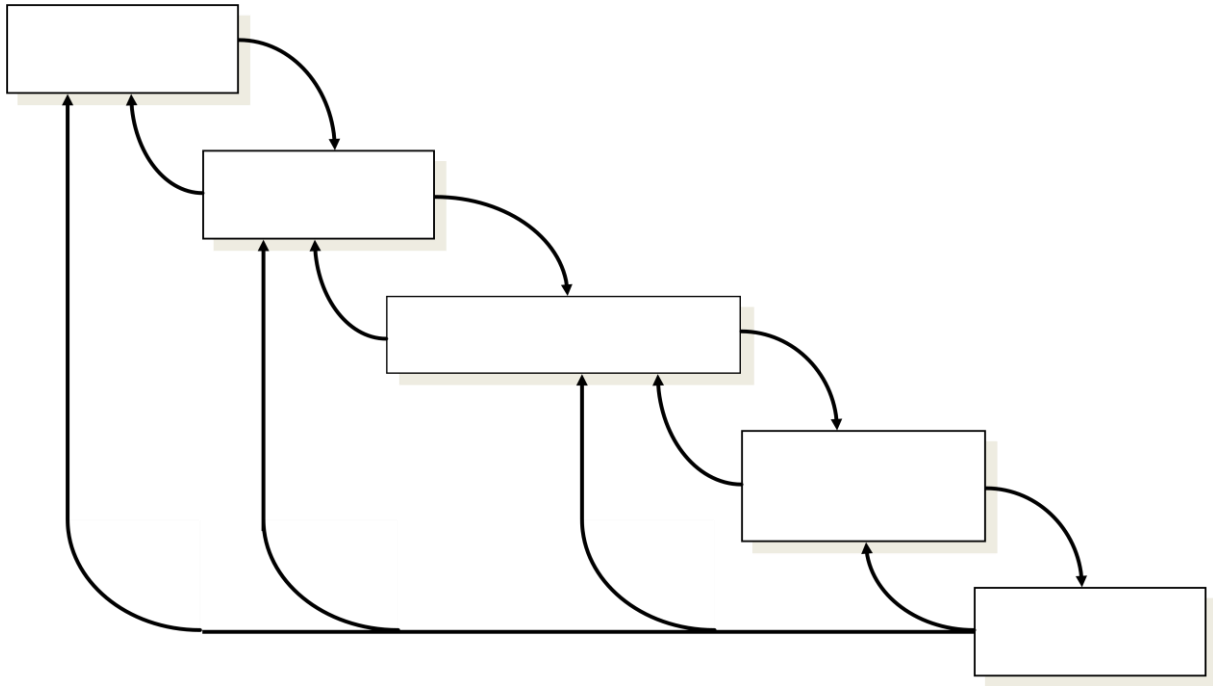
Name:

Matrikelnummer:

## Aufgabe 6 – Softwaretechnik-Allgemeinwissen (15 Punkte)

### Teilaufgabe a)

Tragen Sie in diese Skizze des Wasserfallmodells die Bezeichnungen der typischen Phasen eines Softwareprojektes ein, wie Sie sie in der Vorlesung kennen gelernt haben.



### Teilaufgabe b)

Nennen Sie die in der Vorlesung genannte wesentliche konzeptionelle Schwäche des Wasserfallmodells.

---



---



---



---



---



---

Name:

Matrikelnummer:

### Teilaufgabe c)

Definieren Sie die Design-Prinzipien „hohe Kohäsion“ und „lose Kopplung“ in eigenen Worten.

---

---

---

---

### Teilaufgabe d)

Nennen Sie den Unterschied zwischen einem Framework und einer Bibliothek.

---

---

---

---

---

### Teilaufgabe e)

In welche beiden Arbeitsbereiche unterteilt sich die Entwicklung einer Software-Produktlinie und was wird in diesen beiden Arbeitsbereichen getan?

---

---

---

---

---

---

Name:

Matrikelnummer:

Name:

Matrikelnummer:

Name:

Matrikelnummer: