

**EXAMINATIONS – 2014****TRIMESTER 1****SWEN 223****SOFTWARE ENGINEERING
ANALYSIS****Time Allowed: TWO HOURS****Instructions:**

- Closed Book.
- This examination will be marked out of **120** marks.
- Answer all questions.
- You may answer the questions in any order. Make sure you clearly identify the question you are answering.
- No calculators are permitted.
- Non-electronic Foreign language dictionaries are allowed.

Question	Topic	Marks
1.	Design Principles	20
2.	Object-Oriented Development	20
3.	Software Engineering	20
4.	Class Diagrams and OCL	30
5.	State Diagrams	15
6.	Interaction Diagrams	15
Total		120

Question 1. Design Principles

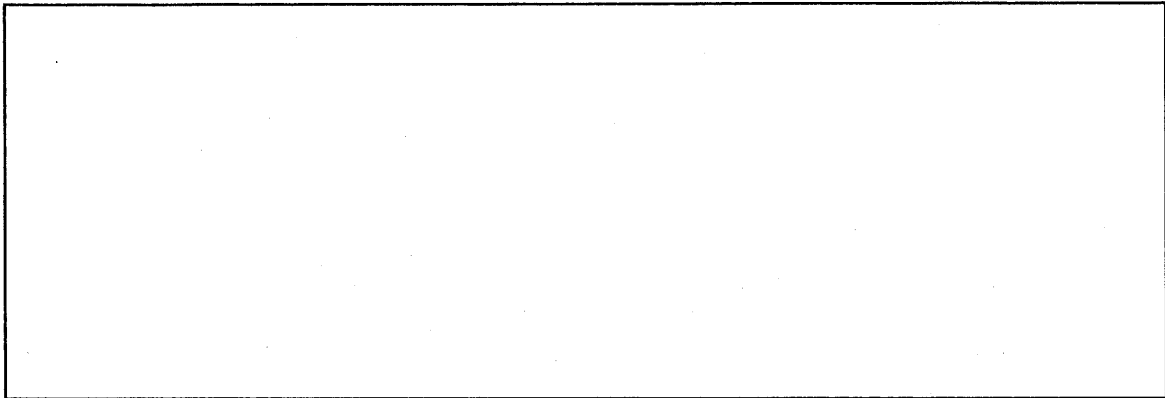
[20 marks]

(a) [5 marks] What is meant by a “System” in Software Engineering? Please describe what it is and give some examples.

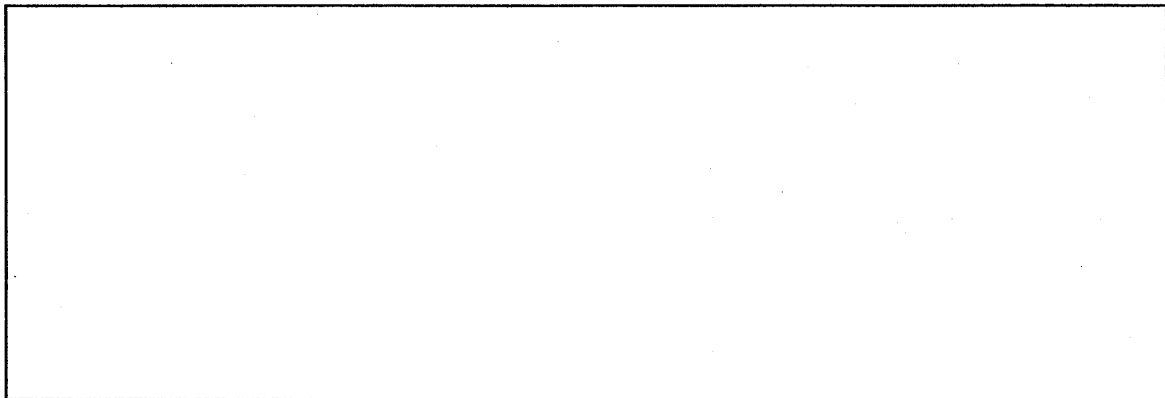
(b) [5 marks] What is a system boundary? Please describe what it is and give some examples.

(Question 1 continued)

(c) [5 marks] Explain the relationship between coupling and information hiding.



(d) [5 marks] So-called *control-objects* manage other, comparatively passive, objects. Discuss the pros and cons of such “manager” objects.



**Question 2. Object-Oriented Development**

[20 marks]

- (a) [5 marks] Briefly explain the purpose and focus for each of the analysis and design phases making sure the difference between the phases is clear.

- (b) [2 marks] Explain the meaning of the term “semantic gap”.

- (c) [3 marks] Suggest how object-oriented software analysis can minimise the semantic gap.

(Question 2 continued)

(d) [5 marks] The UML supports both sequence diagrams and state diagrams. Briefly argue why these are not redundant diagram notations by referring to the different types of behaviour they are intended to capture. Provide one typical example usage for these two diagram notations respectively.

**(Question 2 continued)**

(e) [5 marks] What is the difference between *functional* and *nonfunctional* requirements? Given at least two examples of each.

Question 3. Software Engineering**[20 marks]**

(a) [4 marks] What are the four factors in software development of which the client may prioritize a maximum of three?

(b) [2 marks] Why do you think 40 years of software engineering practice and research have not been able to lead software development out of the above mentioned crisis?

(c) [4 marks] Briefly explain the terms “Correctness” and “Robustness” making sure the difference between them is clear.

(Question 3 continued)

(d) [10 marks] Describe the steps a software engineer can take to minimise the possibility of software project failing using at least two of the examples discussed by at least two different group project presentations in the class.

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

**Question 4. Class Diagrams and OCL**

[30 marks]

(a) [4 marks] List and discuss the properties that make a good **class diagram**.

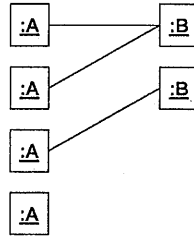
(b) [2 marks] Explain why class diagrams are already created in the **analysis stage** of software engineering.

(Question 4 continued)

(c) [6 marks] **Aggregation** and **composition** are two kinds of **association** in class diagrams. *Give an example for each of them.* Discuss the differences between aggregation and composition. *Use your examples to illustrate the differences.*

(Question 4 continued)

(d) [3 marks] The following figure indicates the **multiplicities** of an association between two classes A and B at the object level. *Draw a corresponding **class diagram**, and show the multiplicities clearly on the diagram.*



(e) [6 marks] Discuss why the use of **association classes** and **higher-arity associations** is a good idea. *Give an example of a **higher-arity association** to illustrate the benefits.*

(f) [3 marks] Explain the purpose of the **Object Constraints Language (OCL)**.

(g) [6 marks] What are **invariants** in the context of OCL? *Give an example of an invariant using OCL and explain its meaning.*

Question 5. State Diagrams

[15 marks]

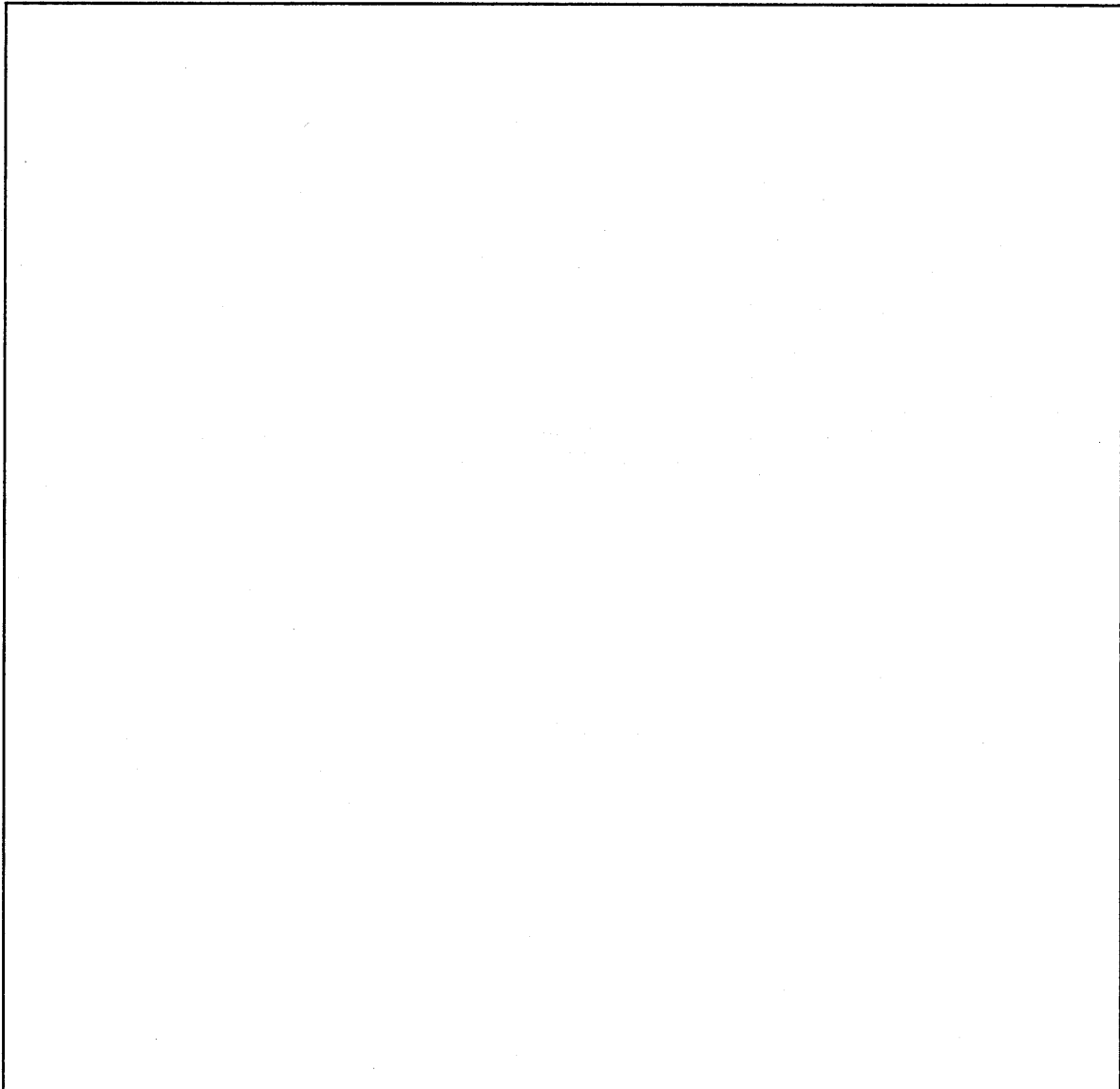
Create a UML **state diagram** that describes the states and events of a phone with the following behaviour:

Initially, the phone is idle. When an incoming call arrives, it keeps ringing until the user picks up or the caller aborts the call. In the former case the phone is connected to the calling party, while in the latter case it becomes idle again.

For an outgoing call, when the user picks up the handle, the phone keeps accepting digits until a valid number has been dialled. When this happens, the phone becomes connected to the called party.

At any point during the dialling or while being connected the user may hang up the phone, causing it to become idle again.

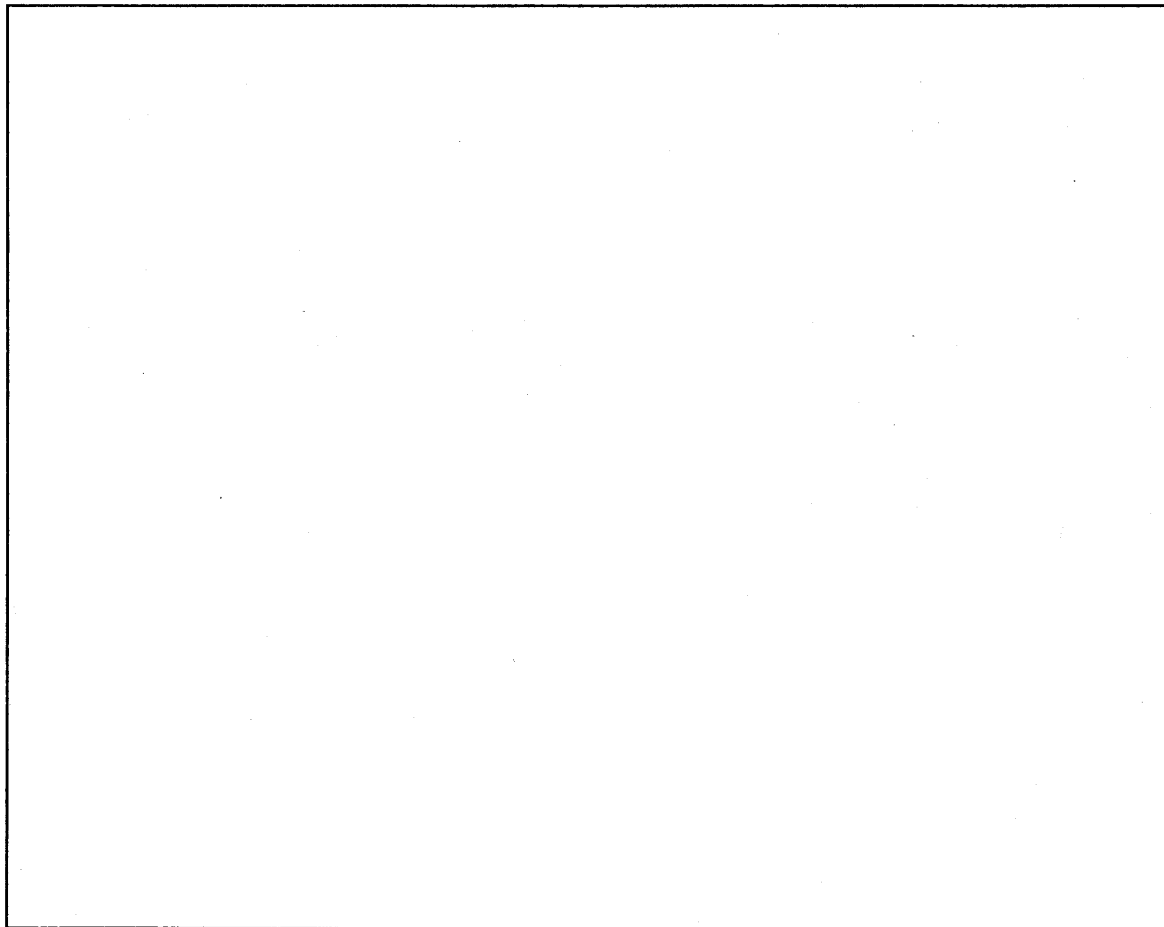
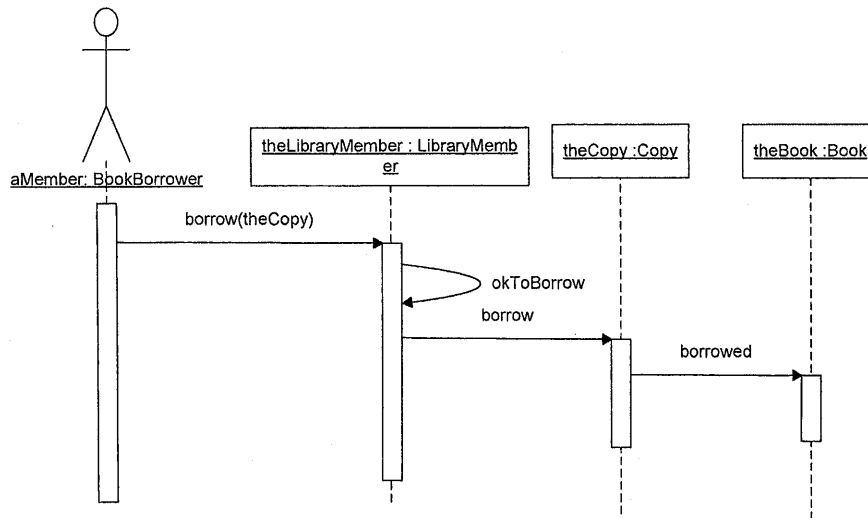
Note: Marks are also awarded for the appropriate use of advanced notation.



Question 6. Interaction Diagrams

[15 marks]

(a) [10 marks] Create a **communication diagram** which contains at least the information in the following sequence diagram:



(Question 6 continued)

(b) [5 marks] Compare and contrast **sequence diagrams** and **communication diagrams**. *Discuss the advantages and disadvantages of both kinds of diagrams.*
