



## BLEKINGE INSTITUTE OF TECHNOLOGY

Written test in (subject): **Basics of Object Oriented Design PA1434**

Date: **May the 29<sup>th</sup>** , 2016

Name: \_\_\_\_\_

Civic number: \_\_\_\_\_

Number of sheets handed in: \_\_\_\_\_

Mark the question(s) you have answered by putting a ring around the relevant number(s)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

### Instructions

A student who cannot produce valid ID will not be permitted to take the examination.

No examination scripts will be accepted by the proctor during the first hour of the examination.

(Students arriving late will thus be permitted to take part in the examination).

Write your name and civic number on each sheet of paper you hand in.

Examination results are posted by e-mail no later than 10 working days after the date of the examination. Exceptions to this rule can occur. In this case, students will be informed by the teacher responsible for the course/program or by the examiner.

All blank answer sheets are to be handed in to the proctor.

### (To be filled in by the proctor)

ID presented: \_\_\_\_\_

Proctor's sign.

Student union fee paid: \_\_\_\_\_

Proctor's sign.

Student union fee not paid: \_\_\_\_\_

Proctor's sign.

### (To be filled in by the teacher )

Number of credits gained: \_\_\_\_\_ Grade: \_\_\_\_\_ ECTS: \_\_\_\_\_ Examiner's sign: \_\_\_\_\_

### (To be filled in and signed by the student, after the correction of the examination)

*I hereby sign my examination script. I am aware that by signing for my script, after correction, I waive my right to contest the examiner's comments and the credits or grade awarded.*

Date \_\_\_\_\_ Signature: \_\_\_\_\_



Ludwik Kuzniarz  
Blekinge Institute of Technology  
School of Computing  
Karlskrona

May the 29<sup>th</sup>, 2016

**Course PA1434**  
**Basics of Object Oriented Design**

**Points**

Question 1	Question 2	Question 3.1	Question 3.2	<b>Question 4</b>	<b>Total</b>

**Grade**

BTH	ECTS

----- Explanations -----

**Questions.**

For the multiple choice questions your task is to indicate the following statements as *true* T or *false* F by placing the appropriate letter indicator in the [ ].

For instance

[ T ] John likes Mary

indicates that the statement is true, or more precisely you think it is true.

John is

[ F ] Swedish

[ T ] English

[ F ] 5 years old

indicates the John is not Swedish, he is English and he is not 5 years old,

If you know that John is German and 20 years old, you should made the following indications:

John is

[ F ] Swedish

[ F ] English

[ F ] 5 years old

For the problem questions your answers should be written in the predefined marked places

either labelled boxes



or along labelled lines .....

Well structured answers will be appreciated.

**Marking**

Every question, just after the question number, has a number of points allocated for that question.

If all entries for the question are marked correctly you obtain that number of points. For any wrong answer for the question one point is subtracted from the number of allocated points but no negative points are generated. It means that if a question has 2 points allocated and has three places to mark T or F then when you make one error you get 1 point for that question, when you make two errors you get 0 points and when all the answers are wrong you also get 0 points.

Test is worth 52 points, 30 point is passed, 42– very good.

**Allowed books**

English – Swedish dictionary

**A remark on drawings**

In the case of tasks that require producing drawings – conceptual models, state diagram, class diagram – please draw first your draft solutions on a spare paper and then redraw them on the marked area on the examination paper trying to arrange the elements (and especially connecting lines) of the picture so that the models were easy readable.

*So, good luck!*

**1. Knowledge****20 p**

## 1. Class

2 p

- ☐ may represent a concept in a domain
- ☐ may represent software element
- ☐ must have attributes
- ☐ must have operations

## 2.

2 p

Sequence Diagram:

- ☐ can be used to describe the activities performed within the use case
- ☐ can be used to describe the activities performed during realization of system operation
- ☐ shows objects, links between objects and messages sent between objects
- ☐ can be replaced by a collaboration diagram

## 3.

2 p

State Diagram

- ☐ can be used to describes the behaviour of a class
- ☐ contains states, events and transitions
- ☐ may be structured
- ☐ may illustrate how attributes are changed in response to events

## 4.

2 p

The tasks performed during the Requirements Analysis include

- ☐ identification of processes of using the system
- ☐ drawing UseCase Diagrams
- ☐ writing operation contracts for the system operations
- ☐ defining the structure of the software system

5. Conceptual Model shows

2 p

- ☐ shows attributes of the concepts
- ☐ shows operations by the concepts
- ☐ relationships between concepts
- ☐ objects and their relationships

6. Observer Pattern suggests a solution for

2 p

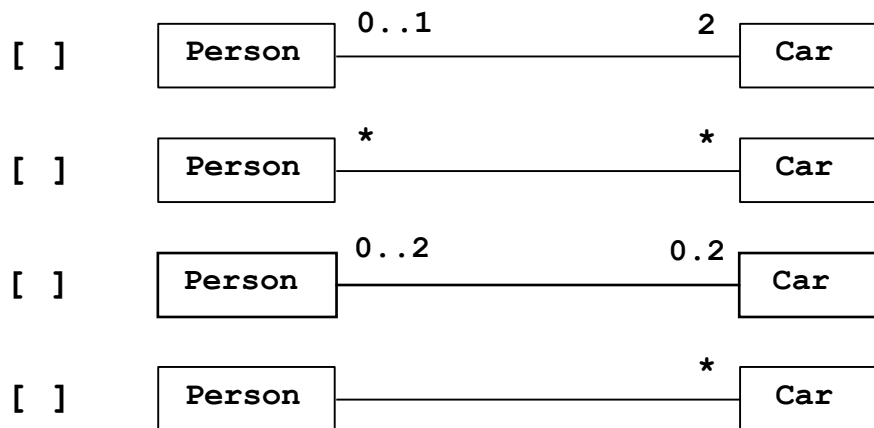
- ☐ keeping cohesion low
- ☐ keeping cohesion high
- ☐ separating user interface and internal information representation
- ☐ managing several views of the same object

7.

3 p

Lars who is a person has two cars : Ford Focus and Opel Vectra. Magnus has no car.

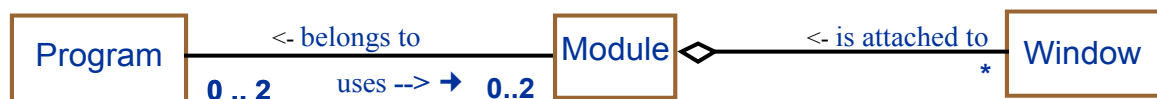
This situation in a domain conforms to (is allowed by) the following conceptual model



8.

5 p

Consider a domain described by the model



Is the situation described bellow consistent with (allowed by) the above model:

- [ ] Orphan:Module does not belong to any Program,
- [ ] There is only one Single:Module belonging to Test :Program,
- [ ] M :Module belongs to Test1:Program and Test2:Program,
- [ ] Game :Program uses two modules GUI:Module and Controller :Module,
- [ ] There must be at least one Module belonging to every Program,
- [ ] A :Module belonging to Strange:Program has no Windows attached to it,
- [ ] The same Module cannot belong to two different Programs,
- [ ] Lonely:Window is not attached to any Module
- [ ] A Common: Window can be removed from M1:Module and attached to M2:Module,
- [ ] Every Window must be attached to a Module.

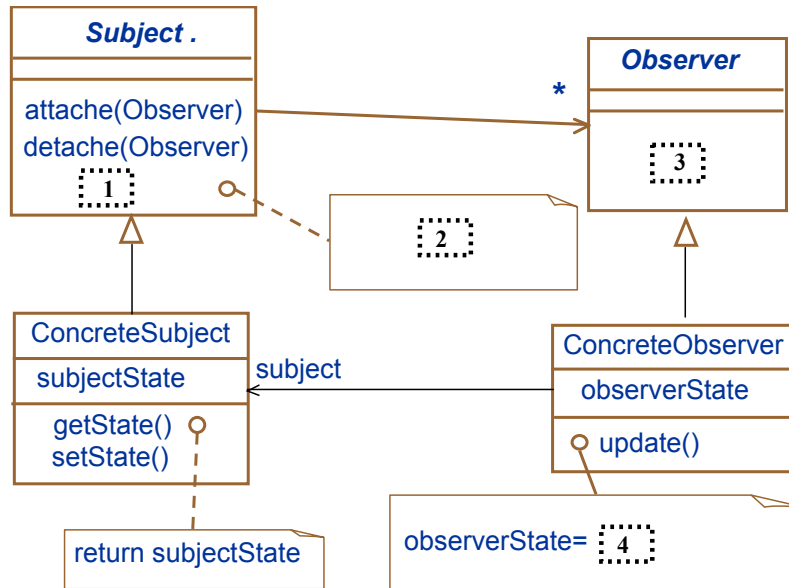
## 2. Designe patterns

### 2.1.

5 p

#### Observer Pattern

A. Complete the pattern – insert missing essential relationships and provide missing operations.



The missing operations are marked with numbered dashed boxes

Provide the contents of the missing parts and the description of the contents of the missing parts:

1. ....  
 ....  
 ....
2. ....  
 ....  
 ....
3. ....  
 ....  
 ....
4. ....  
 ....  
 ....

**3. Skills****33 p****3.1. Object-Oriented Modelling****8 p**

Draw a conceptual model for the following fragment of the problem domain.

In the Country lives a community of persons – workers and students.

Every person has a name and address.

A worker can work in a company or in a school.

A student studies at a school..

Each school consists of from 2 to 5 departments.

Study period at school is four years.

Student has a document – “legitimation” - indicating the faculty and the year she/he is studying.

When employed, each worker has a salary, may be different in different working places.

Workers can work in many companies but student can study only in one school.

At a given moment worker can be unemployed.

Every person can have o vehicle.

There are two sorts of vehicles – bicycles and motorbikes.

Every vehicle has two wheels. A wheel can be small or big.

A motorbike has also an engine. An engine has a label indicting its power – low, normal or high.

An engine is fixed to the motorbike

A worker can have any number of vehicles but student can have only one bicycle.





**3.2. Modelling Behaviour****5**

Car transmission system can be in one of three basic states:

- standing – Neutral gear,
- going backward – Reverse gear and
- going forward – Forward gear.

Driver (user of the system) can push the following buttons:

- R – reverse button,
- F – forward button,
- N – neutral button

(and in consequence generate an appropriate event).

When system is in the state of going forward it can be going on the First, the Second or the Third gear.

When the car is going forward the driver can press **Up** or **Down** button.

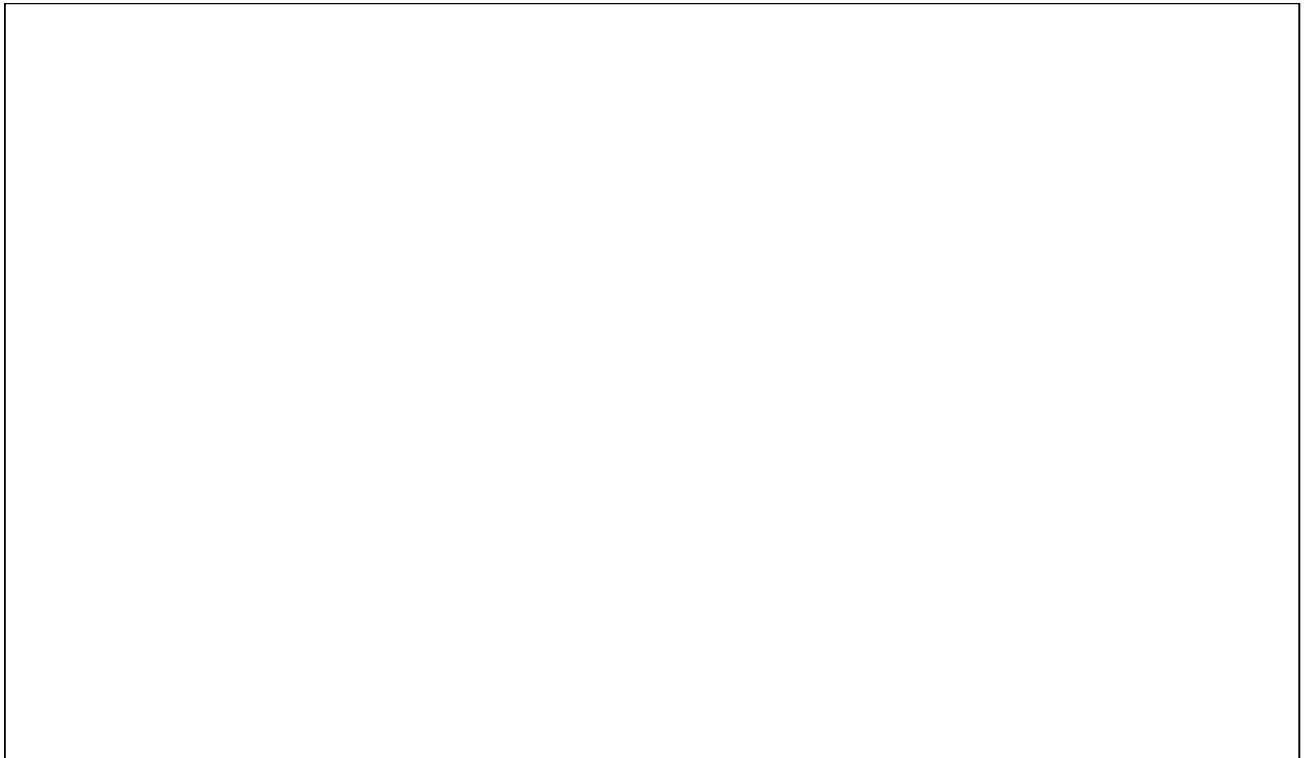
Pressing **Up** button changes the gear one level up – from 1 to 2, from 2 to 3, pressing the **Down** button changes gear from one level down : from 3 to 2, from 2 to 1.

From Neutral driver can go to both states – to going Forward and to going Backward - by pressing the appropriate button.

When going forward the car can at any moment go to the neutral – by pressing neutral.

Changing from neutral to forward sets the transmission system automatically in gear one.

Draw a State Diagram modelling the behaviour of the transmission system.



## 4. Object-Oriented Design

**14 p**

### Cash Machine

Cash Machine is a place from which you can perform a number of operations on your bank account.

It is situated in a number of places around the town and has a direct telecommunication connection with a bank. It has a screen to display information and a number of buttons.

At the Cash Machine you can withdraw money from you account.

When you withdraw money, first before you decide on how much you want in cash, you must check how much money is available on the account, then you input the amount you request, and make a choice on the values of the banknotes (paper money) you want the requested money to be delivered – 100s or 20s (you can either get 100s or 20s). The money you request must be a multiple of 100 (e.g. 200, 500, 1000).

You will not get the money if the requested amount is greater than the amount available on the account

When the terminal is running out of money the terminal maintenance office should be notified.

To withdraw the money and to investigate your account you have to authorize.

For the authorization you have to insert your bank card to a special slot.

A scanner reads the card and asks you to enter PIN code. Then proceeds to the verification process.

During the verification process the machine contacts with the bank, providing appropriate information about the card used for the transaction, and gets a reply if the card is valid.

Then you are asked for a code. If the code is ok you can proceed with the operations otherwise you are asked to retype the code. After three unsuccessful trials your card is kept.

#### 4.1.1. Your tasks

- |   |     |
|---|-----|
| A. Identify actors  | 1 p |
| B. Identify Use Cases   | 1 p |
| C. Draw Use Case Diagram  | 2 p |
| D. Write extended version of <b>WithdrawMoney</b> Use Case<br>Use the Use Case Format introduced on the lectures (...columns, alternative flow ...) | 5 p |
| E. Draw Conceptual Model  | 5 p |

**A. Actors****1 p**

.....	.....
.....	.....
.....	.....
.....	.....

**B. Use cases****1 p**

.....	.....
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....

**C. Use Case Diagram****2 p**

<div></div>
-------------

4 p

[illegible]

E. Conceptual Model = Domain Model

5 p

Domain Model

