Systems Development

(DAT3, SW3, IDA7)

Written exam

5. January 2016, 12:30-16:30

Name	
Study line	
Student number (**not** CPR number)	

The exam set consists of 13 pages (including this page). There are two problems: Problem 1 has 5 sub-problems, and problem 2 has 3 sub-problems. The weight of each problem is given.

You have 4 hours to complete the exam.

Write your solution to each assignment in the space provided in this set. Only solutions written in the set can be handed in. Make a draft of your solution before filling in the solution. Obtain draft paper from the exam officials.

Use a readable handwriting in your solution.

The following exam aids are permitted: The textbook, copies of slides and other course material, and personal notes from the course.

Personal notes etc. may be brought in printed form or stored on computers or tablets. Note, however, that you may not use computers, tablets, or other communication devices to communicate with other persons or search for information and assistance from the outside. Cell phones are *not* permitted.

You must provide your student id upon request from the officials.

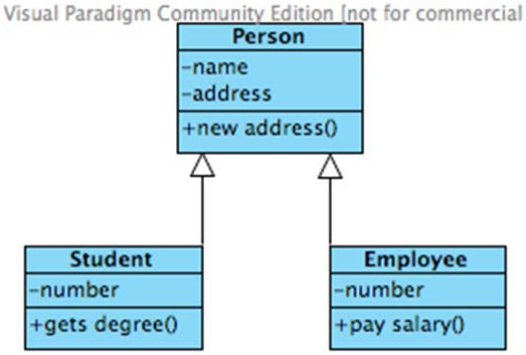
Problem 1. A university system (60%)

Consider a system for a university where we want to keep track of students and employees. We know about students that they have names, addresses and student identification numbers. We know about employees that they have names, addresses and employee identification numbers. None of the identification numbers is the social security number (CPR number).

1.1 (10%)

a) (5%)Describe what class diagram 1 is modelling in terms of classes and structures.

b) (5%)Suppose that we have a student named Maren Turi who lives at Ved Stranden 4 and has the student identification number 12345. We also have an employee named Jens Bang at Østerågade 9 with the employee identification number 12346. In the system modeled in class diagram 1, how many objects are there and what values describe their state?



Class diagram 1

Answer 1.1

a) It's a generalization structure where 'Person' is the general class and 'Student' and 'Employee' are the specific classes. The 'Student' and 'Employee' both have all the properties that 'Person' has.
b) There'll be two objects. The student object has the values "Maren Turi", "Ved Stranden 4", and "12345". The employee object has the values "Jens Bang", "Østerå 9" and "12346".

1.2 (5%) Suppose now that Maren Turi gets hired by the university as a part-time programmer and that Jens Bang also begins studying at the university. In the system modeled in class diagram 1, how many objects are there and what values describe their state?

Answer 1.2

There'll be four objects:
A student object with the values "Maren Turi", "Ved Stranden 4", and "12345".

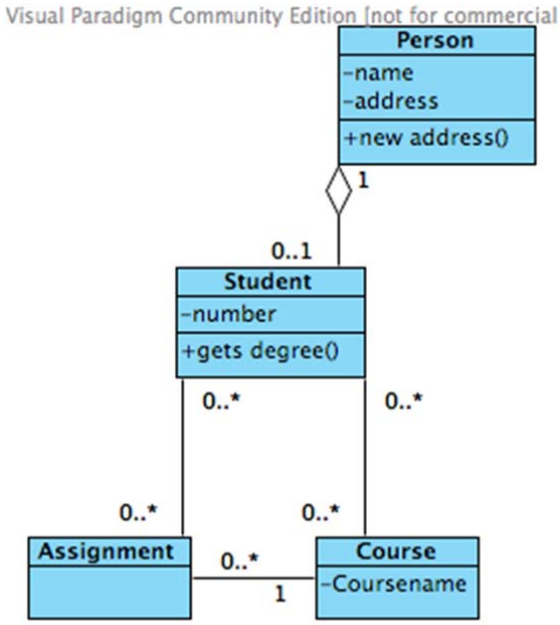
An employee object with the values "Jens Bang", "Østerågade 9" and "12346".

An employee object with the values "Maren Turi", "Ved Stranden 4", and an employee number we don't know.

A student object with the values "Jens Bang", "Østerågade 9" and a student number we don't know.

Class diagram 2 and event table 1 show classes, structures and events in the part of the problem domain concerned with students, courses and assignments (The Employee class and associated events have been omitted).

Note: Students cannot register to a course *after* teaching has begun and assignments cannot be published *before* teaching begins.



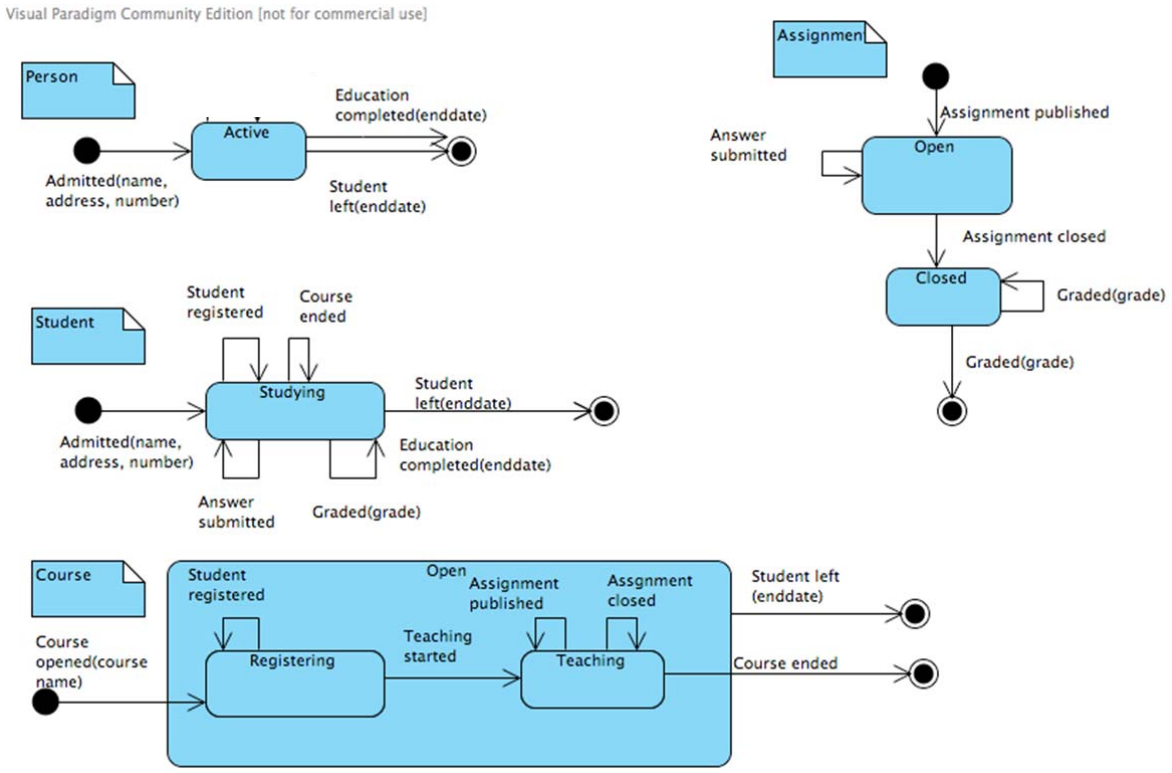
Class diagram 2

	Person	Student	Course	Assignment
Admitted(name, address, number)	+	+		
Education completed (enddate)	+	+		
Student left (enddate)	+	+	*	
Student registered		*	*	
Course opened(coursename)			+	
Teaching started			+	
Course ended		*	+	
Assignment published			*	+
Assignment closed			*	+
Answer submitted		*		*
Graded(grade)		*		*

Event table 1

1.3 (20%) Make behavioral patterns for the classes in class diagram 2. Include event attributes where relevant

Answer to 1.3



Answer to 1.3 (cont'd)

1.4 (20%) Complete the activity 'model component' for the model in problem 1.3. Write the resulting class diagram in the space below. Remember to include all attributes.

Answer 1.4

New classes:

Answer and *Grade* as a relationship between Student and Assignment. *Grade* can be omitted, but must then be added as an attribute to class Answer

Registration as a relationship between Student and Course

Attributes

Class	**Attributes (new and old)**
Person	name, personstate
Student	number, enddate, studentstatus
Course	coursename, coursestatus
Assignment	assignmentstate
Answer	grade, answerstate (if Grade and Answer classes are merged)

The university has the policy that a student who fails more than 50% of the assignments for more than 3 courses in a semester will receive a warning.

The system developers have, among others, defined the following functions: *register grade*, *send warning*, *find average*, *register student*, *see assignment*, *show grade*, *submit answer* and *end study*

1.5 (5%) What are the types of these functions?

	Read	Compute	Update	Signal
register grade			X	
send warning				X
find average		X		
register student			X	
see assignment	X			
show grade	X			
submit answer			X	
end study			X	

Problem 2. A sailing school (40%)

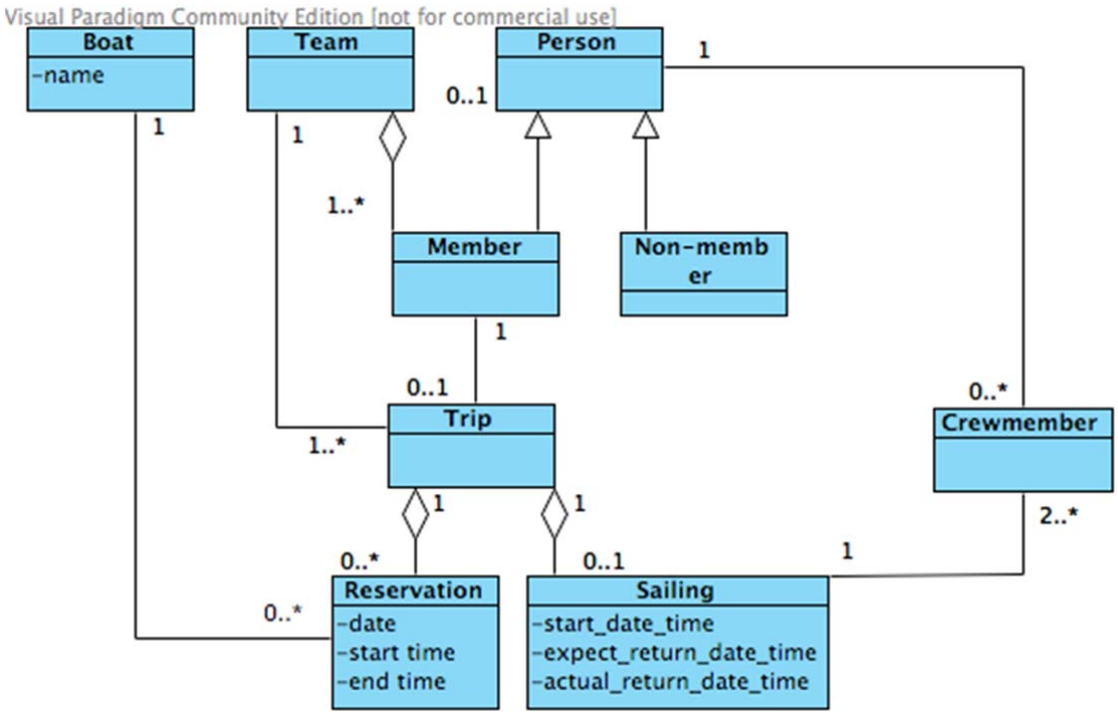
A sailing school in a sailing club uses a number of boats for instruction. Other members of the club can use a boat for a private trip when it is not being used by the school. The club has asked a group of system developers to build an IT based system to help them with administering reservations and trips.

The system developers have studied the problem domain and learnt that:

- Reservations must be made prior to a trip.
- Reservations are for a specific boat. A boat has a name.
- A reservation includes the date, start and end times for the intended trip.
- A sailing school team consists of a number of club members.
- A team reserves a number of trips for the whole summer; e.g.; boat Aramis every Tuesday from 18-22.
- Members can also make reservations, but only for one trip at a time.
- There must be at least two persons in the crew on a trip.
- The following is registered for a trip: start_date_time, expected_return_date_time, actual_return_date_time.
- Prior to leaving the harbor, all persons in the crew must be registered. Note that this applies to sailing school teams as well; i.e. there may be guests on the trip or some in the team may not be present.
- Persons in a crew can be both members and non-members of the club. Non-members cannot make reservations, however.
- Note: There are a set of complex rules regulating the composition of a crew. The systems developers have decided to implement these rules in the functions, and you can disregard them in your answer to the following questions.

2.1 (25%) Make a class diagram for the sailing school based on the above information.

Answer to 2.1



Note: The *Reservation* and *Sailing* classes can be omitted in which case their attributes must be moved to the *Trip* class.

The crewmember class can (or should?) be associated with the Sailing class as well.

The analysis of the application domain resulted in the following information

- **Actors**: members, non-members, and sailing school manager
- **Use-cases**: reserve trip, cancel reservation, create school team, reserve for school team, register trip start, assign member to crew, assign non-member to crew, register trip end.

2.2 (5%) Fill in the actor table below, based on the information you now have about the sailing school

Actors / Use cases	Sailing school manager	Member	Non-Member
Reserve trip		X	
Cancel reservation		X	
Create school team	X		
Reserve for school team	X		
Register trip start		X	
Assign member to crew		X	
Assign non-member to crew			X
Register trip end		X	

For the design, the developers decide upon a combination of a closed-strict generic architecture and a client-server architecture with local presentation. There will be two types of clients. A mobile client from which members can make and cancel reservations and a client on a computer in the sailing club, which has access to all functions.

2.3 (10%). Draw the component architecture for the system. You can disregard the technical platform and communication/cryptography components.

Answer, problem 2.3