Software Design

Final exam

January 30th 2018.

During this Exam I will not undertake any illegal acts of accepting or providing any solutions to other students. I also state that my health condition is good and that I am capable for taking this Exam.

Student ID	*Name and Surname*	*Hand-written signature*

1. (1 point) Name at least four types of software projects.

2. (1 point) Define the software process.

3. (1 point) List requirements elicitation methods.

4. (1 point) Describe keypoints of *Unified Process* model.

5. (1 point) Name the main characteristics of Scum teams organisation.

6. (1 point) Describe software design principle called *design defensively*.

7. (2 points) The following source code describes the Class `BankService` and interface `IQueue`:

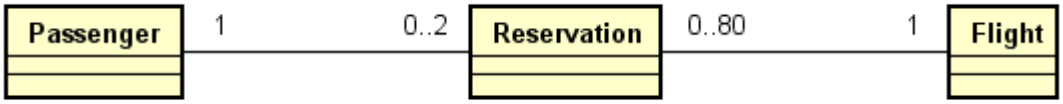
```
/**
 * Class BankService
 */
public abstract class BankService implements IQueue {

    private int numberInQueue = 0;

    public static final int MAX_QUEUE = 10;

    public abstract void service (int [] params );

    public boolean addLast()
    {
        if (numberInQueue == BankService.MAX_QUEUE) return false;
        numberInQueue = numberInQueue + 1;
        Print("new in queue, now "+numberInQueue);  //console print method
        return true;
    }

    public boolean removeFirst()
    {
        if (numberInQueue == 0) return false;
        numberInQueue = numberInQueue - 1;
        Ispis("first one left, remaining "+numberInQueue);  // console print method
        return true;
    }
    ...
}


/**
 * Interface IQueue
 */
public interface IQueue {

    public boolean addLast ();

    public boolean removeFirst ();

}
```

a) (1 point) Is the following line of code in the `main()` method correct? Explain.

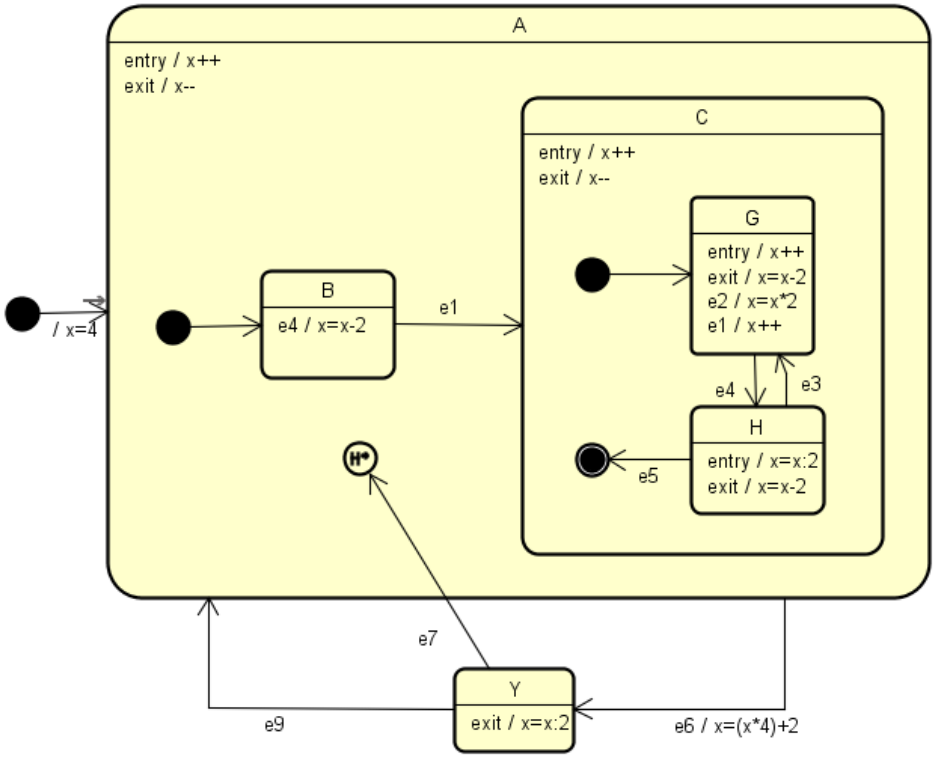
```
IQueue queue = new BankService();
```

b) (1 point) Does the class `BankService` implement interface `IQueue correctly`? Explain.

8. (1 point) According to the UML diagram below answer the following: what is the minimum and what is the maximum number of reservations if there are 5 passengers and 2 flights present in the system?



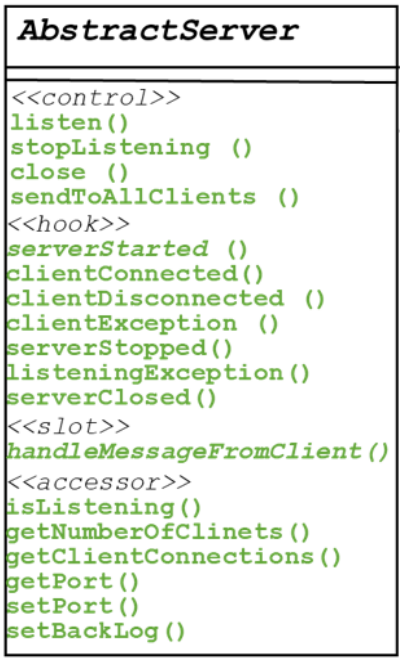
8. (2 points) For the given UML Statemachine diagram determine the value of variable x upon the execution of events: **e1, e2, e2, e6, e7** (**x = 4** at the beginning). List all actions!



9. (1 point) Draw a UML Component Diagram for two components that are connected through an interface. Name and explain in short component interface types.

10. (1 point) Describe the basic Model-View-Controller (MVC) architectural pattern.

11. (1 point) Which method group(s) in the presented AbstractServer class, according to named stereotypes, can but do not need to be implemented by a software developer. What is the purpose of this group of methods in framework environment?



12. (1 point) What is the main goal of software testing?

13. (2 points) For functional testing using equivalence partitioning:
    a) (1 point) List steps of testing process.
    b) (1 point) For an example of 4-digit number from the interval [5000, 9998], write down minimal test cases.

14. (2 points) For the following procedure:
    a) (1 point) Draw a *control flow graph*.
    b) (1 point) Determine the upper test case number limit which guarantees full code coverage. Write the formula and calculate the result.

```
01. insertion_procedure (int a[], int p [], int N)
02.  {
03.      int i,j,k;
04.      for (i=0;i<=N; i++)
05.          p[i] = i;
06.      for (i=2;i<=N; i++)
07.        {
08.         k=p[i];j=1;
09.         while (a[p[j-1]] > a[k]) {
10.           p[j] = p[j-1];
11.           j--
12.           }
13.          p[j] = k;
14.        }
15.  }
```

15. (1 point) Describe the property of tiers as defined in the strict *n-tier* architecture.

16. (1 point) As a member of a software development team during the semester you work with GitLab version control system for project development. Where was the data which is edited by each team member stored? Was it possible for multiple members to work simultaneously on the same source code file?

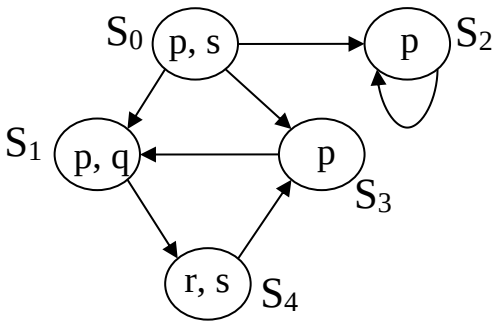
17. (1 point) Define necessary predicates and constants and map the sentences into well-defined formulas of first-order logic:

    "On a faculty there is only one elective course called ‘Expert systems’".

18. (1 point) Translate the following sentence from natural language to CTL (Computational Tree Logic) formula:

    "After req=1 there is always ack=1 at the end."

19. (1 point) For the given model of Kripke structure M, determine set of all states for which the CTL formula **A** (p **U** r) is valid.



Problem solving **– Parking lot machine**

Parking lot machines sell parking tickets. Machine displays basic information like date and time and lists all credit cards types accepted for paying the parking ticket.
User who wants to buy a parking ticket must first select one of the credit card types displayed on screen. Then the machine opens the credit card input slot and waits for user to insert the credit card into it. Once the user has inserted the credit card, the number of parking hours can be set. The default number of hours is "1" but user can increase that number by pressing "+" button, or decrease it by pressing "-" button.
At any point before the purchase is confirmed, user can cancel the purchase and machine returns the credit card.
Once the user confirms the purchase, the credit card payment transaction is carried out[1]. If the transaction has been successful the machine prints out the parking ticket. If the transaction has failed for any reason an appropriate message is displayed on screen. Simultaneously with printing ticket or displaying a failed transaction message, the machine returns the credit card.

At the Parking System Headquarters there is a server computer which runs a message server, a database server and a web server. Each parking lot machine has an embedded GSM communication module which is used to periodically send a message about machine status to Headquarters message server via XMPP/TLS protocol.
The message server runs an application which handles incoming messages from parking machines and stores data in MySQL database located on the database server. The web server runs a web application which communicates with database and allows system administrator to access the system remotely via web browser over a secure HTTPs link.

21. (4 points) Statemachine diagram

Model the user interaction with parking lot machine using a UML Statemachine diagram.

22. (4 points) Activity diagram

Model the parking ticket purchase process using UML activity diagram form the moment the user selects the credit card type. Also note that in order to carry out the transaction, parking machine connects to server at Parking System Headquarters which handles the transaction with bank. Disregard the possibility of user cancelling the purchase.

23. (4 points) Deployment diagram

Use UML instance level deployment diagram to model the system with 3 parking lot machines which communicate with server at Parking System Headquarters. Choose meaningful instance names and indicate appropriate stereotypes.

[1] NOTE: No PIN is required for transaction.