Computer and Systems
Engineering Department
Faculty of Engineering
Alexandria University

Answer All Questions (Total 90 marks):

Q1)

a) **"Favor Composition over Inheritance" is one of the design principles, explain why and give an example.** *(5 marks)*

b) **There are two principal implementation approaches for the Observer pattern. In the first approach, all Observers are automatically notified whenever the Subject experiences any state changes and each Observer determines whether or not to retrieve any of the new state information from the Subject. In the second approach, the Subject automatically transmits all state-change information to every Observer. Discuss the relative advantages of each approach, providing an example for which each approach would be appropriate.** *(5 marks)*

c) **Draw class diagram for an immutable object. What are the pros and cons of using immutable objects?** *(5 marks)*

Q2)

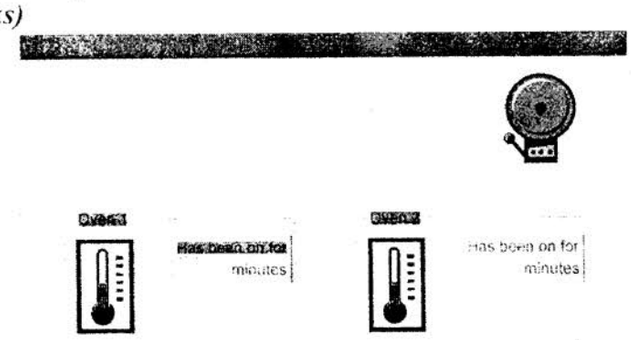
a) In the approach to "analysis" and "design" that has been studied in this course, briefly compare and contrast these two terms. What is "analysis" about? What is "design" about? *(5 marks)*

b) Consider a (simplified) monitoring application whose GUI is depicted in the figure below. The application is used in a bakery to control two ovens. This application receives events from each oven (turned on/off, temperature change) which it displays on the thermometers shown. In case an oven is off, its group of GUI elements must be grayed out. The monitoring application must keep timers in order to fire an alarm whenever any of the ovens has been on without interruption for more than 15 min at a temperature of 250°C or higher. The monitoring application itself sends no signals to the ovens, it lacks all control functions.

(i) Prepare the class diagram for the ovens and the monitoring application *(5 marks)*

(ii) Prepare the statechart for the monitoring application to update the GUI. *(5 marks)*

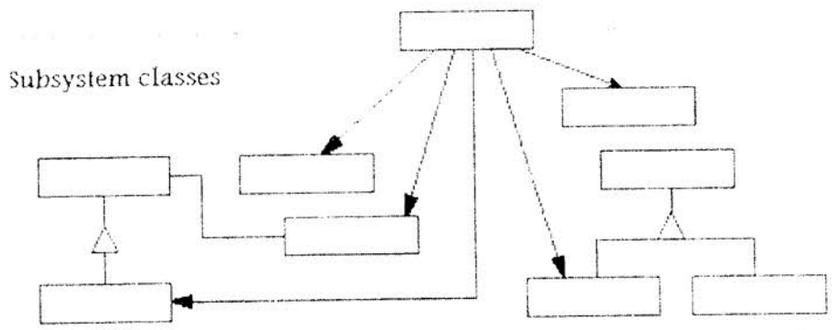
(iii) Specify and motivate your choices of three design patterns that can be used in this application. *(5 marks)*



Q3)

a) **Notice that the State pattern and the Strategy pattern have exactly the same class diagram. Explain, then, the fundamental difference between these two patterns with respect to their intent** *(5 marks)*

b) **Consider the system diagram below. Which design pattern do you recognize here? When should this design pattern be applied?** *(5 marks)*



c) **What is the difference between coupling and cohesion? What are the goals of good software design regarding the coupling and cohesion degree? Give an example for a good coupling level and an example of a good cohesion level.** *(5 marks)*

Q4)

a) **When a stream of bytes is flowing to or from a network socket, it is frequently desirable to have the Socket class extended to allow outgoing bytes to be encrypted and incoming bytes to be decrypted. Because of legal restrictions on the import and export of encryption software, this new EncryptedSocket class needs to be independent of the classes used to implement the encryption algorithms (DES, RSA, etc.). Explain why the Factory Method Pattern is appropriate for this situation and draw a UML class diagram illustrating its application here.** *(10 marks)*

b) **A Tetris field is essentially a big collection of stones, some representing empty fields, and some representing parts of a Tetris block, either just falling or landed at the bottom of the play ground some time ago. Whenever a block lands, it dissembles into its individual stones, which are considered independent from then on. It makes sense, therefore, to represent each stone by an individual object in an object-oriented Tetris application. Doing so naively results in a large amount of objects being allocated, deleted, and moved all the time.**

What design pattern can be applied to optimize this, taking advantage of the fact that the stones are really very similar? Motivate your answer and draw the design pattern structure/class diagram. *(10 marks)*

Q5)

a) **A forum is a website that provides the following functionality: i) start a conversation thread, ii) comment (post a message) on a specific conversation thread, iii) register to a specific conversation thread to get updates by email each time someone posts a message.**

Questions:

i. **How would you implement such a system using Composite and Observer design patterns? Consider that the Observer class is extended with a class Mailer that sends the emails to the registered users when a new post is added. Sketch the UML diagram of the system.** *(5 marks)*

ii. **How would you add new functionality that sends SMS messages with the posts added? Extend the UML diagram from i).** *(5 marks)*

b) **Opening a new connection to a Database Management system is usually an expensive operation. Most of the systems keep of a pool of available connections.**

i. **Draw the Object Pool design pattern structure and explain its participants.** *(5 marks)*

ii. **Write enough code fragments to show the implementation of the Pool Design Pattern to keep the released unneeded connections for future usage.** *(5 marks)*