Faculty of Engineering, Alexandria University	Year: 2nd year
Object-Oriented Programming	Duration: 180 Minutes
Dept. of Computer and Systems Engineering	Final Exam: 2008

Read the following scenario CAREFULLY then answer ALL SIX questions:

A retail shop wishes to build an internet presence. The shop manages the products in separate catalogs. Each catalog contains many products and each product may appear in one catalog only. The catalog has a name, a validity date range and a supplier name. The shop classifies the products using one catalog grouping system (CGS). A CGS is a tree of product categories (similar to the file explorer in MS Windows). Each product is assigned to one node. A node normally contains lots of products. The internal nodes have a description and contain other nodes (similar to the directories in the files explorer). Unlike directories, internal nodes do not have products. Leaf nodes (nodes without children) contain only products. A product is identified through a unique identifier. It contains a short and a long description in all languages available in the shop. The price information is given in EGP only but is subject to discounts according to the ordered quantity. All price levels must be modeled in the application. The languages supported in the system are defined by the administrator. The end user sees the description of the product and nodes in the same language as the locale setting in the browser.

The user can search for products by typing in a search string or by browsing the CGS. The user selects products from the result set. The selected products are added to the shopping cart. The user can edit the quantities of each product in the shopping cart. The total price is dynamically calculated depending on the quantities ordered. The user then proceeds to the checkout area. In this area, the user supplies the shipping address and the credit card information.

In order to control the management of the products in the shop. Each catalog is imported first in a staging area. In this area, the shop administrators can edit the product information, its prices, etc. but the product is not visible to the user. After a publish operation on the catalog, the content are made visible to the user. After publishing, the administrator can block one or more products for short periods (e.g., if the product is out of stock). No editing is allowed here. In this case, the user finds the product but cannot add it to the shopping cart.

Question 1 (10%):

Construct a simple use case diagram involving the administrator, the catalog, the product, the shopping cart and the user.

Question 2 (20%):

Draw the UML state diagram for a product.
Transform it to a JAVA class skeleton using the *state* design pattern.

Question 3 (20%):

Draw the UML class diagram for the system. All attributes mentioned in the scenario must be present in the diagram. All operations needed in Questions 1 and 2 as well as any obvious methods in the problem description must be included in the diagram. Make all necessary assumptions.

Question 4 (20%):

Map the UML class Diagram of Question 3 into a code **skeleton** written in JAVA. All associations between classes must be present in the code. Make all necessary assumptions.

Question 5 (20%):

In the result set screen, it is possible that the result of the user query returns more than 1000 products. Loading them in memory for display takes a long time. The web page displays only 20 products per page. Usually the user does not go beyond the second page. Suggest an efficient solution based on one (or a combination) of the design patterns studied in the course. Identify the design pattern(s) used, draw the UML class diagram and write a JAVA code skeleton.

Question 6 (10%):

a. Draw the UML class diagram of the *singleton* design pattern
b. Draw the UML class diagram of the *push filter* design pattern.

GOOD LUCK