CITY UNIVERSITY OF HONG KONG

Course code & title : CS3342 Software Design

Session : Semester A 2013/14

Time allowed : Two hours

This paper has 17 pages (including this cover page).

---

1. This paper consists of 6 questions.
2. Answer **ALL** questions.
3. Write your answers in the space provided under each question.

---

*This is a **closed-book** examination.*

No materials or aids are allowed during the whole examination. If any unauthorized materials or aids are found on a candidate during the examination, the candidate will be subject to disciplinary action.

Student Number: _____

Programme: _____

Seat Number: _____

Question	1	2	3	4	5	6	Total
Max(%)	15	10	20	30	5	20	100

Question 1: Software Engineering and OO Methodologies (15 Marks) (CILO #1)

(a) Draw a *Waterfall* software development model, and explain at least three disadvantages of applying this model in software development. (5 Marks)

(b) Provide explanations as to what are an **Object** and a **Class** in terms of Object Oriented Software Design, and what are their **relationships** in between? You may provide examples to aid your explanations (10 Marks)

Question 2: Software Ethics (10 Marks) – CILO5

Danny is a software programmer working within a large team for the development of a new Internet Browser Application. The next release date (deadline) of the new application is tomorrow, but Danny knows that the coding tasks he is responsible has a bug not being addressed, he knows the bug would only cause the application to crash in 1 out of 1,000 chances and may not be severe enough to delay the next release date. So he decides not to report this issue to his senior manager because further testing and debugging takes time and may jeopardize his reputation in the work environment.

Actually *Danny* has the following options:

1. Keep silent about the potential issue, on the grounds that there is only 1 in a 1,000 chances that the bug would causing problem.
2. Keep silent about the potential issue, since it would only cause embarrassment and his reputation in the work environment.
3. Tell the project manager about this issue.

What would you do, if you were Danny? Explain and justify **your** decision. You may wish to use the 5P process to justify your decision (Purpose, Pride, Patience, Persistence, Perspective)

I am Danny, I will _____

Based on the consideration of ethics in software development, use the 5P evaluation process to analyze the ethics and justify your decision in the following:

Purpose: What will you do in selecting the best decision? What are the consequences?

>

Pride: Would you feel pride in keeping the known problem as a secret, which may cause major problem later on to the project if it fails to run properly?

>

Patience: Would you set aside a time, talk to someone whose judgment that you can trust and think carefully?

>

Persistence: Have you tried to analyze all solutions to resolve either keeping the secret or telling the manager?

>

Perspective: Even if you feel your judgment will not be affected, how will it set a good example to other colleagues?

>

Question 3: Requirements Elicitation and Analysis (20 Marks) – CILO2

(a) Study the following. Draw a **use case diagram** of a *Ticket Vending Machine* system for MetroTrain. (10 Marks)

A Customer arrives at the train station ticket vending machine:

1. She has 3 options (use cases), to buy: *One-way ticket*, *Weekly-pass*, *Monthly-pass*.
2. In each of these three options, the system must be able to handle abnormal operational situations (i.e. when there is an operational error, handle with the *ExceptionHandling* use case)
3. She can choose multiple items. After all the selections of above are completed, she may proceed to *CheckOut*, which (1) it will *CalculateTotal* amount, and then (2) proceed to the *Payment* screen, the *Payment* will be validated by an external system called *PaymentValidationSystem*.
4. Only when the *Payment* transaction is successfully completed, then it will issue all the purchased tickets at once via *IssueTickets*.
5. The customer can now continue her journey with ticket(s) purchased.

Whenever possible, your use case diagram MUST use <<Extend>> or <<Include>> to provide a good use case diagram.

(b) Requirements Specifications (10 Marks)

Based on the same case study described in (3a), complete the following table to describe the **Checkout** use case under typical course of events, and alternative course of events. The situation involves the *Customer* actor, as well as an external payment processing system called *PaymentValidationSystem*

Use Case Name:	**Checkout**	
Actor(s):	Customer, PaymentValidationSystem	
Description:	This use case describes the process of a customer to completing the checkout procedure for the tickets selected. On successful completion, tickets will be issued.	
Reference ID:	METRO-1.0	
Typical course of events:	**Actor Action**	**System Response**
Alternative course of events:		
Precondition:	Checkout can only be made after at least one ticket is selected to purchase.	
Postcondition:	The completed transactions will be recorded.	

Question 4: Object Oriented Analysis and Modeling (30 Marks) - CILO3

(a) Object Oriented Class Diagram (20 Marks)

The *Banana Software Company* is one of the leading software development firms in the world. With employees exceeding 20,000 to work on over 1,000 software development projects, they need a new system to manage different kind of employees (staffs) and be able to allocate them to specific projects.

The company needs to keep each *Employee*'s:

- *employeeID*, *lastName*, *firstName*, *DOB*(date of birth) and *gender*, and be able to compute their pay using an operation called *computePay()*.

Banana employs both *Full-time* and *Part-time* employees:

- *Full-time* employees has a record of their *annualSalary* and a operation *computeTax()* to workout their annual tax commitments.
- *Part-time* employees are normally be paid by a pre-agreed *hourlyWage*, therefore the operation to compute their pay is different to other employees, which is using *computePay(hours)*.

There are many *Employee*s as well as there are many *Project*s owned by the company:

- Each *Project* has a *projectID, projectName, startDate* and *endDate*.
- *Employee*s are allocated/assigned to *Project*s (we called this *ProjectStaffAllocation* in the system), more importantly their <u>*roles*</u> in the *Project*s are <u>different</u> and needs to be recorded. For example Employee Miss Ada can be allocated to Project A as Project Manager (as a *role*), and she can also be allocated to Project B as Test Engineer (as a *role*).

Develop a complete class diagram to show the above scenario using what you have learned and best practices in object oriented software design. Multiplicity and class relationships must be shown correctly in the diagram.

Class Diagram for the Banana Software Company described above:

(b) Object Oriented Sequence Diagram (10 Marks)

DairyFarmer supermarket Checkout sequence:

1. **Customer** (Actor) can choose a large variety of groceries and products before selected products are taken to the **Cashier** (Object) system for checkout.
2. After received selected product items from the customer, the **Cashier** (Object) will first create a new **Order** (Object).
3. In the newly created **Order** (Object), it will search for the product price of the first scanned item using **Product_Item** (Object). Once the price is received, **Cashier** (Object) will be able to add the item quantity and price to the **Order** (Object).
4. After a confirmation to add an item from the **Order** (Object) is received for the current item, repeat step 3 for the next item until all products are added to Order successfully.
5. After all items are added into the order, **Cashier** (Object) will request the current **Order** (Object) to calculate the order total.
6. After the order total is received, it will be forwarded back and displayed to the **Customer**. End.

Your task is to illustrate the above steps using a sequence diagram, it may be containing combined fragments. *(Hint: potential lifeline columns: Customer(Actor), Cashier, Order, and Product_Item)*

Question 5: Roles of Variables (5 Marks) – CILO 4

(Different *Roles of Variables*)

Role	Description
Constant/ Fixed value	A variable which is initialized without any calculation and whose value does not change thereafter.
Stepper	A variable stepping through values that can be predicted as soon as the succession starts.
Most-recent holder	A variable holding the latest value encountered in going through a succession of values.
Most-wanted holder	A variable holding the "best" value encountered so far in going through a succession of values. There are no restrictions on how to measure the goodness of a value.
Gatherer	A variable accumulating the effect of individual values in going through a succession of values.
Transformation	A variable that always gets its new value from the same calculation from value(s)of other variable(s).
Follower	A variable that gets its values by following another variable.
One-way flag	A two-valued variable that cannot get its initial value once its value has been changed.
Temporary	A variable holding some value for a very short time only.
Organizer	An array which is only used for rearranging its elements after initialization.
Other	Any other variable.

Study the following pseudo codes and identify the role of each variable declared in the code listing by completing the table below (you may use the reference table above):

```
public int RandomTotal (int how_many)
  {
     int max = 10;
     int sum = 0;
     int [] rands_array = new int [how_many];
     Random random = new Random();

     for (int i=0; i<how_many; i++)
     {
        rands_array[i] = random.nextInt(max);
        sum = sum + rands_array[i];
     }
     return sum;
  }
```

Variable	Role	5 Marks
how_many		(1 Mark)
max		(1 Mark)
sum		(1 Mark)
rands_array		(1 Mark)
i		(1 Mark)
random	Other	----------

Question 6: Object Oriented Design Principles (20 Marks) – CILO 4

Case Study: Space Invader Game Design (Using OCP Principle)

The following is a simple implementation of the popular game "Space Invader". In the game, **AlienShip** and **HumanShip** will fight each other using a default gun **Weapon** that causes only 10% of damage to the opponent. Within the code below, the following containing an design issue which disallows the game to operate a more powerful **weapon** such as **Laser** or **Nuke,** which will cause more damages to the opponent in the game.

Your first task is to study the following pseudo code and analysis the problem.

```
class SpaceShip
{
    private int life;
    public void attack (SpaceShip ship)
    {
        Weapon p = new Weapon();
        if (ship.isAlive())
            p.Fire(ship);
    }
    public void takeDamage (int damage) {
        if ((life - damage) > 0)
            life = life - damage;
        else
            life = 0;
    }
    public boolean isAlive(){
        if (life > 0)
            return true;
        else
            return false;
    }
}
```

```
class HumanShip extends SpaceShip
{
    public HumanShip()
    {
    }
}
```

```
class AlienShip extends SpaceShip
{
    public AlienShip()
    {
    }
}
```

```
class Weapon
{
    public void Fire (SpaceShip target) {
        target.takeDamage(10);
    }
}
```

```
class Laser extends Weapon
{
    public void Fire (SpaceShip target)
    {
        super.Fire(target);
        target.takeDamage(20);
    }
}
```

```
class Nuke extends Weapon
{
    public void Fire (SpaceShip target)
    {
        super.Fire(target);
        target.takeDamage(50);
    }
}
```

```
class GameController
{
    public static void main(String[] args){
        //Creation of an alien spaceship and a human spaceship.
      SpaceShip a_ship = new AlienShip();
      SpaceShip h_ship = new HumanShip();
        //The human spaceship is under attack by the alien spaceship.
      a_ship.attack (h_ship);
        //The human spaceship is firing back to the alien spaceship.
      h_ship.attack (a_ship);
    }
}
```

(a) Based on your understanding of the pseudo code above, draw a simple class diagram to illustrate the class interactions and inheritance. You do not need to show attributes and operations. **(5 Marks)**

(b) Based on the OCP design principle, modify the codes in the application, so that **AlienShip** and **HumanShip** can operate more powerful weapons (Laser or Nuke) than only the default **Weapon** to fight each other. (You ONLY need to show and indicate the code fragments /parts that are needed to modify)

New Requirements: You need to modify the main game control sequence, so that specifically:
1. Allow the **AlienShip** to first use the **Laser** weapon to attack the **HumanShip**.
2. And then to allow the **HumanShip** to launch the **Nuke** Weapon (Nuclear) to attack the **AlienShip**.

In the following, show your modified codes and provide explanations to your solution. **(10 Marks)**

(c) What if we want to include a new **Weapon** called **Missile,** which takes additional 35 points of damages than that of the standard **Weapon**, how and where would you put it in the design? **(5 Marks)**

-- END --