

INSTITUTE OF TECHNOLOGY BLANCHARDSTOWN



**NATIONAL DIPLOMA IN COMPUTING
(Information Technology)**

**Object Orientation with Design Patterns
Module code here**

Semester I

Internal Examiner(s): Stephen Sheridan

**External Examiner(s): Dr Gerard Parr,
Dr Micheal O'hEigeartaigh**

**August 2002
Time of examination here**

Instructions to candidates:

- 1) Section A: Five parts to be attempted. Section B: Answer any 3 Questions.**
- 2) All questions carry equal marks.**

**DO NOT TURN OVER THIS PAGE UNTIL YOU
ARE TOLD TO DO SO**

(Section A: Answer any 5 parts of Question 1)

Question 1

- a) What are the overall goals of the Design Pattern approach to software development? In particular how does it differ from more traditional methods?

[5 Marks]

- b) Describe the difference between *Class Adapters* and *Object Adapters*.

[5 Marks]

- c) Any object in Java can be queried to see what methods and parameters make up its interface. Write a code sample that demonstrates how you would query a Java object for its methods only.

[5 Marks]

- d) Briefly describe how the *Proxy Design Pattern* works and give three situations where it might be used.

[5 Marks]

- e) Name *five* Behavioural Design Patterns.

[5 Marks]

- f) Discuss the following code sample with regard to *FlyWeight* classes:

```
String s1 = new String("Hello");  
String s2 = new String("Hello");  
  
if (s1 == s2)  
    System.out.println("Same");
```

[5 Marks]

- g) What are *Concrete Methods*?

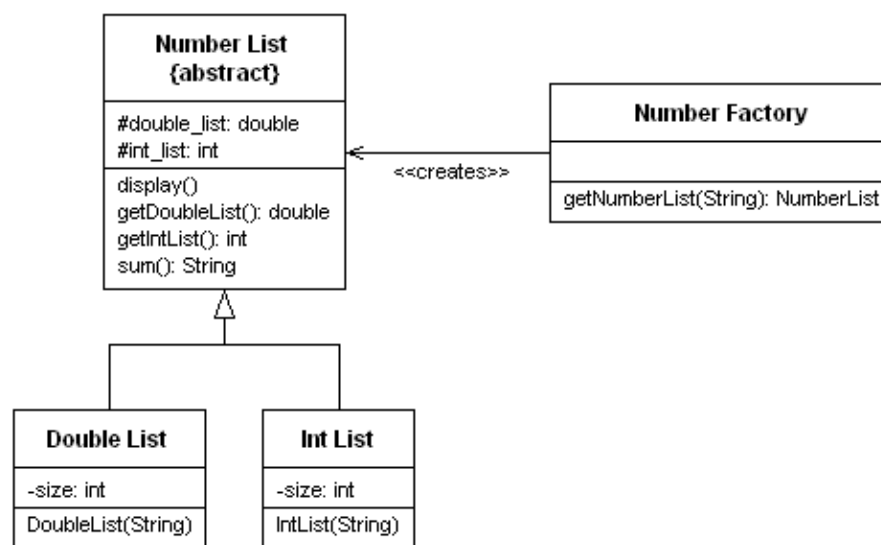
[5 Marks]

[Total Marks 25]

(Section B : 3 Questions to be Attempted)

Question 2

- a) What are Creational Patterns? Name four. [6 Marks]
- b) What is the *Singleton Pattern* and when would you use it? Write a simple Singleton Class that uses exception handling to force the programmer to deal with the possibility of null pointers. [8 Marks]
- c) The following UML class diagram describes a Simple Factory Pattern that is used to create two different kinds of *NumberList* objects based on a value passed to the *getNumberList* method.



Using the following test code as a reference, implement each of the classes shown in the UML class diagram above.

```
String list1 = new String("1 2 3 4 5 6 7 8 9 10");
String list2 = new String("1.1 2.2 3.3 4.4 5.5 6.6
                          7.7 8.8 9.9 10.1");
```

```
NumberFactory nfactory = new NumberFactory();
nfactory.getNumberList(list1).display();
```

```
NumberList numberlist2 =
    nfactory.getNumberList(list2);
```

```
numberlist2.display();
```

```
System.out.println("Sum of list 2 : " +
    numberlist2.sum());
```

[11 Marks]
[Total Marks 25]

Question 3

- a) With the aid of some sample code describe the differences between *Object Adapters* and *Class Adapters*.

[6 Marks]

- b) The program given in **code listing 1** uses a simple *Decorator Pattern* to create a *FileFilter*. A *FileFilter* converts the first character of every sentence it reads to an uppercase character. The output of the program is shown below.

```
*** Output using standard FileInputStream ***
hi guys. i thought i'd send a brief note. this email
system is very hard to use. i have to use lowercase.

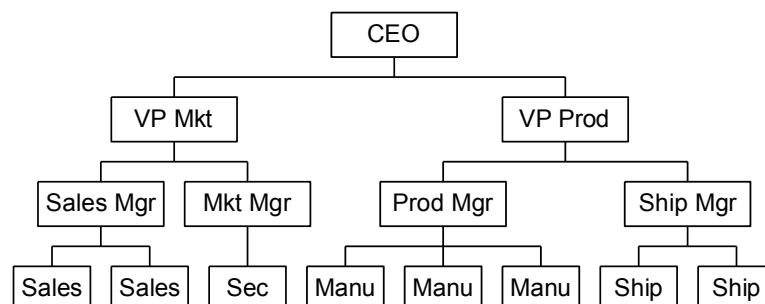
*** Output using FileFilter ***
Hi guys. I thought i'd send a brief note. This email
system is very hard to use. I have to use lowercase.
```

Describe in detail the role each class plays in order to implement the *Decorator Pattern* along with an explanation of each of the classes methods.

[8 Marks]

- c) The organisation chart for a small company is shown below. If the company is successful each of the company members receives a salary, and we could at any time be asked for the cost of the control span of any employee. We define this control span as the salary of that person as well as the combined salaries of all of his or her subordinates. Therefore:

- The cost of an individual employee is that employees salary
- The cost of an employee who heads a department is that employees salary plus the salaries of all employees that the employee controls.



We would like a single interface for calculating the salary totals correctly regardless of whether or not the employee has subordinates:

```
public float getSalaries();
```

With the aid of code samples describe how you would go about solving this problem using the *Composite Design Pattern*.

[11 Marks]

[Total Marks 25]

Question 4

- a) The *Chain of Responsibility Pattern* does not have to use a linear chain. What does this statement mean? What, if any, implications are there if a non-linear chain is used?

[4 Marks]

- b) When you build a Java user interface, you provide controls – menu items, buttons, check boxes, and so on – to allow the user to tell the program what to do. When a user selects one of these controls, the program receives an *ActionEvent* which it must trap by implementing the *ActionListener* interfaces (*actionPerformed*). This code can get quite cumbersome if there are many controls that make up the user interface.

Describe with the aid of some sample code one method that can be used to reduce the amount of coding in the *actionPerformed* method by forwarding specific commands to specific user interface controls.

[10 Marks]

- c) What is the Model View Controller Architecture (MVC) and what are its overall goals? With the aid of code samples describe one case in Java where the MVC architecture is used.

[11 Marks]

[Total Marks 25]

Question 5

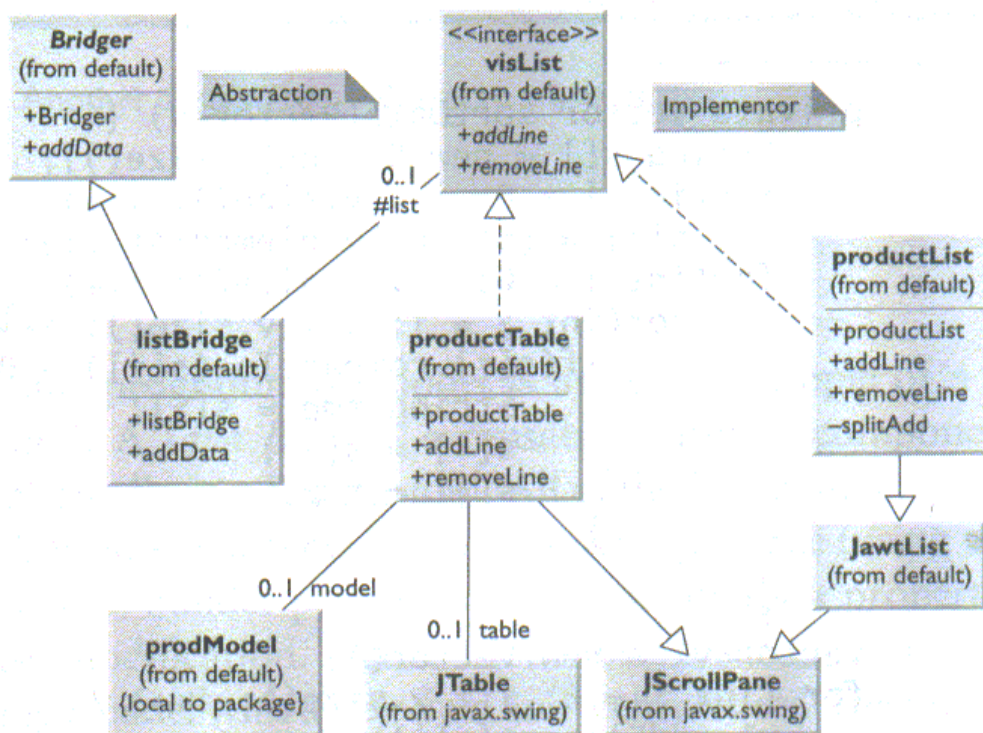
- a) The following code sample is taken from a Behavioural Design Pattern. Name the pattern and briefly describe how the two interfaces are used.

```
public interface Observer
{
    public void sendNotify(String s);
}

public interface Subject
{
    public void registerInterest(Observer obs);
}
```

[6 Marks]

- b) The following UML class diagram describes a commonly used creational pattern. Name the pattern and describe how each of the classes in the diagram contributes to the patterns implementation.



[8 Marks]

Given a Vector containing a list of names write a class which implements the Enumeration interface so as to provide a filter that will only iterate through names that contain a given substring.

[11 Marks]

[Total Marks 25]