# C:\Documents and Settings\hylim\Local Settings\Temporary Internet Files\OLKE6\Nilai Uni.jpg

**BACHELOR IN SOFTWARE ENGINEERING (HONS)**

Assignment

January 2023

Module: EC3307/EC3375: Object & Component Technology/Software Design

Pattern and Technology

Lecturer: Ms. R.Rajesvary

Instructions:

## Case Evaluation (Individual Assignment)

Start Date:

Due Date:

Weightage: 20% of the total assessment for the course

Total mark is 100.

This question paper consists of 3 printed pages

(excluding front cover)

**Case Evaluation Report**

**Description:**

**The overall aims of this assignment is to give you the opportunity to practice selecting and using design patterns for a specific application.**

**You are required you to analyse the problem below, examine the alternative solutions, and propose the most effective solution using supportive evidence.**

**Problem: A Graphical System**

You are required to design a system that can represent and render (draw on the screen) graphical objects composed of multiple points, lines, ellipses, rectangles and triangles. An example of the kind of graphical object your system may need to represent and render is given below:

All graphical objects in your system must provide the following methods (we will not concern ourselves with the details of how to implement each of these methods):

public void Draw(Window w)

Renders the object in window w.

public void Translate(int x\_trans, int y\_trans )

The value of x\_trans is added to the x coordintate(s) of the object. The value of y\_trans is added to the y coordinate9(s) of the object.

public void Scale(int s)

The x and y coordinate(s) of the object are multiplied by s.

public void Rotate(Point p, double thisAngle)

The object is rotated about point p by thisAngle.

The client of your system wants to be able to treat individual components (e.g. points, lines, ellipses, rectangles and triangles) and graphics consisting of compositions of components (e.g. other polygons, the house shown above, etc.) uniformly. In other words, they want to be able to apply any of the operations (i.e. draw, translate, scale and rotate) to composite graphics in the same way that they could be applied to the individual graphic objects. So, for example, if the client had created a graphic corresponding to the house diagram above, then applying a rotation of 90 degrees about centre of the house should end up with the graphic shown below (note that the client calls the rotate command on the entire graphic, your system will take care of how to rotate each of the components of the graphic):

**Task 1**

Decide which design pattern to use to design a representation of the objects for the graphics system. Justify your choice. Using this design pattern, draw the class diagram for this representation. Identify the attributes and methods to be defined in each class. Define the method implementations that are suggested by the design pattern.

### Task 2

### A client of your system creates the following composite graphic objects for the house example:

They first create an object which represents the door and the handle (composed of an ellipse, a rectangle and a triangle):

Next, they create an object which represents the front of the house and the windows:

Then they add the object which represents the door to the object which represents the front of the house:

Finally, they create an object which is composed of the house front object and a triangle representing the roof:

Draw a diagram of the final set of objects and their references for the house graphic.

**Task 3**

An additional feature is required of the graphics system that will allow the client to display a bounding box around an existing individual or composite graphic object:

The method:

private void boundingBox(graphicObject)

will calculate where the bounding box of the object referenced by graphicObject should be and will draw it. Note that all the operations which can be applied to graphics objects (i.e. draw, translate, scale and rotate) should also apply to any bounding boxes that are around any object within the graphic. So if in the previous example the client had put a bounding box around the house graphic, then a subsequent application of the rotation method should also rotate the bounding box:

Choose a design pattern with which to implement the bounding box feature of the graphic system. Justify your choice. Redraw your class hierarchy, including any new classes for the bounding boxes which are required by the design pattern. Identify the attributes and methods to be defined in each class and outline the method implementations that are suggested by the design pattern you have used. Note that one of the new classes should implement the boundingBox(...) method.

**Task 4**

Suppose that the client decided to put a bounding box around the house graphic which was created in Task 2. Redraw the set of objects and their references after the bounding box has been added.

**Submission guideline:**

Your report should contain the following:

* Standard cover page
* Table of contents (with page numbers for easy reference)
* Introduction
* Task 1
  + Design pattern and justification
  + Class diagram with attributes and methods
  + Method implementations (code or pseudocode)
* Task 2
  + A diagram of the final set of objects and their references for the house graphic.
* Task 3
  + Design pattern with which to implement the bounding box feature of the graphic system and justification.
  + Class diagram with attributes and methods
  + Method implementations (code or pseudocode)
* Task 4
  + A diagram of the set of objects and their references after the bounding box has been added.
* Conclusion

Marking Scheme

Task 1 30%

Design pattern and justification 5%

Class diagram with attributes and methods 10%

Method implementations (code or pseudocode) 15%

Task 2 15%

Task 3 30%

Design pattern and justification 5%

Class diagram with attributes and methods 10%

Method implementations (code or pseudocode) 15%

Task 4 15%

Overall documentation 10%

**Assignment - Marking Scheme**

**Name:**

**ID:**

|  |  |  |  |
| --- | --- | --- | --- |
| Marking Criteria | Marks Allocation | Marks Obtained | Comments |
| Task I  1) Design Pattern and Justification  2) Class Diagram  3) Pseudocode | 5%  10%  15% |  |  |
| Task II  Composite diagram | 15% |  |  |
| Task III  1) Design Pattern and Justification  2) Class Diagram  3) Pseudocode | 5%  10%  15% |  |  |
| Task IV  Composite diagram | 15% |  |  |
| Documentation | 10% |  |  |
| Total | 100 |  |  |