SET A

UNIVERSITI KUALA LUMPUR
Malaysia France Institute

FINAL EXAMINATION
JANUARY 2014 SESSION

SUBJECT CODE	:	**FSB23804**
SUBJECT TITLE	:	**OBJECT ORIENTED PROGRAMMING & DATA STRUCTURE**
LEVEL	:	**BACHELOR**
TIME / DURATION	:	**X.XX pm – X.XX pm (3 HOURS)**
DATE	:	

INSTRUCTIONS TO CANDIDATES

1. **Please read the instructions given in the question paper CAREFULLY.**

2. **This question paper is printed on both sides of the paper.**

3. **Please write your answers on the answer booklet provided.**

4. **Answer should be written in blue or black ink except for sketching, graphic and illustration.**

5. **This question paper consists of TWO (2) sections. Section A and B. Answer all questions in Section A. For Section B, answer two (2) questions only.**

6. **Answer all questions in English.**

THERE ARE 8 PAGES OF QUESTIONS, EXCLUDING THIS PAGE.

SECTION A (Total: 40 marks)

INSTRUCTION: Answer ALL questions.
Please use the answer booklet provided.

QUESTION 1

(a)   Describe the **software life cycle**.

(4 marks)

(b)   **Design** is one of the steps during the **development phase** of **software life cycle**. Identify the two design techniques used during this step.

(2 marks)

(c)   **Inheritance** and **polymorphism** are the processes that characterize Object Oriented Programming (OOP). **Define** the both terms.

(4 marks)

QUESTION 2

Given the following class:

```
class Rectangle
   {
      protected double length;
      protected double width;
      public Rectangle(double l, double w)
       {
         length = l;
         width = w;
       }
      public double setLength(double length)
       {
         return length;
       }
      public double setLength(double width)
       {
         return width;
       }
      public double GetArea()
```

```
      {
       return length * width;
      }
    public void Display()
      {
        Console.WriteLine("Length: {0}", length);
        Console.WriteLine("Width: {0}", width);
        Console.WriteLine("Area: {0}", GetArea());
      }
  }//end class Rectangle

class Tabletop : Rectangle
  {
    private double cost;
    public Tabletop(double l, double w) : base(l, w)
      { }
    public double GetCost()
      {
        double cost;
        cost = GetArea() * 70;
        return cost;
      }
    public void Display()
      {
        base.display();
        Console.WriteLine("Cost: {0}", GetCost());
      }
  }
```

(a)   Illustrate the above classes' relationship using **Unified Modeling Language (UML).**

(4 marks)

(b)   Based on the code segment given below, give the output of the program.

```
  static void Main(string[] args)
    {
       Tabletop t = new Tabletop(4.5, 7.5);
       t.Display();
       Console.ReadLine();
    }
```

(2 marks)

(c)   Modify the code in **Main** so that the program will prompt user to enter the value of the length and width.

(4 marks)

QUESTION 3

Given the declaration of a linked list:

```
LinkedList<double> myll = new LinkedList<double>();
```

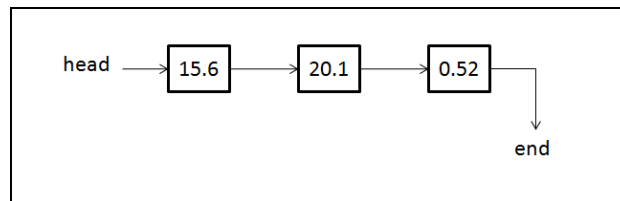(a) Write a segment of code to create the linked list shown in Figure 1:

Figure 1: linked list

(3 marks)

(b) Find the node that contains the value of 20.1 and put it in a new created
LinkedListNode named myNode.

(1 mark)

(c) Insert a new node which contains the value of 3.5 after myNode.

(1 mark)

(d) Check the existence of the node that contains the value of 0.52. If it exists, make
head points to it.

(5 marks)

QUESTION 4

Given the following declaration of stack:

```
Stack<string> myStack = new Stack<string>();
```

(a) Elaborate the concept of stack.

(2 marks)

(b) Write the statements for the instructions below:

     i.   Add the values given below into the stack:

Paris – Nice – Lille – Montpellier

(2 marks)

     ii.   Display the top value in the stack.

(1 mark)

     iii.   Remove the top two values of the stack.

(1 mark)

     iv.   Display all the elements of the stack from the top to the lowest.

(4 mark)

SECTION B (Total: 60 marks)

INSTRUCTION: Answer only TWO questions.
Please use the answer booklet provided.

QUESTION 5

As a software engineer, you are under a software development team in a company X. Your team is given a task to create software which is used to **store the information of the company projects**. A meeting has been done to discuss the task and your team came up with the idea to break your team members into three (3) sections which are:
- Graphical User Interface (GUI)
- Database
- Core Program

You are assigned as a leader for **Core Program** section. In this section, you have discussed with all your section partners (3 people) about the possible solution. All of you then agreed on the following details:

*"A project should have an **id**, a **title**, a **description** of the project, a **supervisor** and **team members"***

In such case, your section has decided to create three classes which are `Employee`, `EmployeeList` and `Project`. The class `Employee` will be used to create supervisor, `EmployeeList` will be used to store a list of team members and `Project` will regroup all the elements from the phrase above. Since your section just has three (3) members, you have decided then to assign a class for a member. You are in charge of the class `Project`.

Your other two (2) members have come up with this UML designs:

```
              Employee
-----------------------------------
- id : string
- name : string
-----------------------------------
+ Employee()
+ Employee(string,string)
+ GetID() : string
+ GetName() : string
+ SetID(string) : void
+ SetName(string) : void
```

Figure 2: UML Design of Employee

```
┌──────────────────────────────────┐
│           EmployeeList           │
├──────────────────────────────────┤
│ -maxSize : int                   │
│ - currentSize : int              │
│ - list : Employee[]              │
├──────────────────────────────────┤
│ + EmployeeList()                 │
│ + EmployeeList(int maxSize)      │
│ + Count() : int                  │
│ + ElementAt(int) : Employee      │
│ + Add(Employee) : void           │
│ + Insert(Employee, int) : void   │
│ + Retrieve(Employee) : int       │
│ + Remove(Employee) : void        │
└──────────────────────────────────┘
```
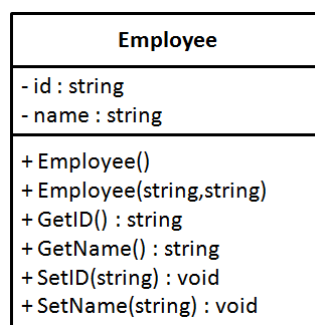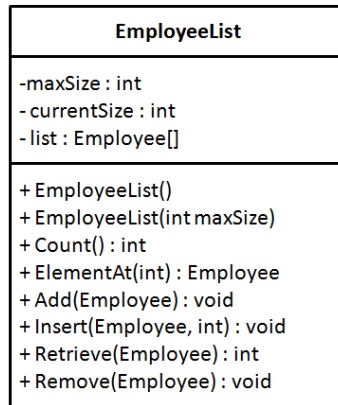
Figure 3: UML Design of EmployeeList

(a) Sketch UML design for the class `Project` based on the phrase in the double quotes ("…") above and the following information:

- The class should have a **constructor** with **three (3)** `string` **parameters** which are used to initialize the **id, title** and the **description** of a project.

- The class should be able to **assign** a supervisor.

- The class should be able to **add** a new member or **remove** a member.

- The class should contain a method to **display** all the information of the project.

(10 marks)

(b) Based on UML design sketched in (a), Figure 2 and Figure 3, write the program for the class `Project`.

(20 marks)

QUESTION 6

You are developing an **inventory system** for a **plywood factory**. You have been given the following information to start your program:

-   Each plywood has its own unique serial number.
-   A plywood is characterized by its length, width and thickness. They are stored and localized according to these characteristics.
-   While placing or arranging the plywood into a stack, there is maximum number of plywood that must be respected.

You are at the beginning of the development phase. You are planning to create two classes, Plywood and Plywood Inventory. You have come up with the UML designs of both classes which are as the following:
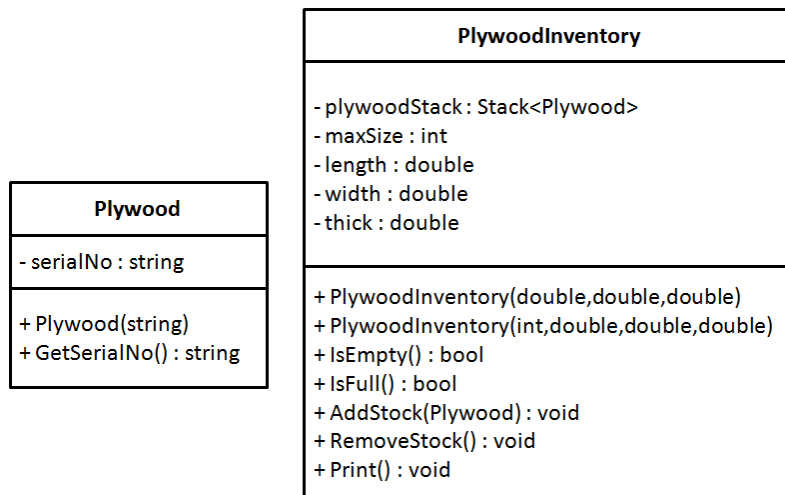
```
                                   ┌───────────────────────────────────────────┐
                                   │             PlywoodInventory                │
                                   ├───────────────────────────────────────────┤
                                   │ - plywoodStack : Stack<Plywood>             │
                                   │ - maxSize : int                             │
                                   │ - length : double                           │
                                   │ - width : double                            │
                                   │ - thick : double                            │
           ┌─────────────────────┐ ├───────────────────────────────────────────┤
           │       Plywood        │ │ + PlywoodInventory(double,double,double)    │
           ├─────────────────────┤ │ + PlywoodInventory(int,double,double,double)│
           │ - serialNo : string  │ │ + IsEmpty() : bool                          │
           ├─────────────────────┤ │ + IsFull() : bool                           │
           │ + Plywood(string)    │ │ + AddStock(Plywood) : void                  │
           │ + GetSerialNo() : string│ + RemoveStock() : void                    │
           └─────────────────────┘ │ + Print() : void                            │
                                   └───────────────────────────────────────────┘
```

Figure 4: UML Designs for `Plywood` and `PlywoodInventory` Classes

Plywood

It has a serial number, a constructor to initialize the serial number and a method which returns the value of serial number.

PlywoodInventory

It has a stack of plywood, maximum number of plywood that a stack can contain, the length, width and thickness of the plywood in the stack, two types of constructor, the methods to check if the stack is empty or full, a method to add a new stock, a method to remove a stock and a method to display all the information of the `PlywoodInventory` object.

   (a) Create the class `Plywood`

                                                                        (4 marks)

   (b) Create the class `PlywoodInventory`

                                                                        (26 marks)

QUESTION 7

(a) Given the following list:

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
16	30	24	7	62	45	5	55

Figure 5: List of 8 integers

Briefly explain the concept of sorting by sketching the initial unsorted list in Figure 5 towards getting the sorted list using the following methods:

i.   Selection sort

(8 marks)

ii.   Insertion sort.

(8 marks)

(b) Briefly explain the concept of searching the value of x in a list using the following methods:

i.   Sequential search.

(7 marks)

ii.   Binary search.

(7 marks)

END OF QUESTION