SET A

UNIVERSITI KUALA LUMPUR
Malaysia France Institute

FINAL EXAMINATION

SEPTEMBER 2013 SESSION

SUBJECT CODE	:	FSB23103
SUBJECT TITLE	:	OBJECT ORIENTED PROGRAMMING
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 11 PAGES OF QUESTIONS, EXCLUDING THIS PAGE.

SECTION A (Total: 40 marks)

INSTRUCTION: Answer ALL questions.

Please use the answer booklet provided.

Question 1

Answer the following questions about the **four (4)** major phases in **software life cycle.**

(a) Briefly explain about the phase called **Development**.

(4 marks)

(b) Give **two (2)** reasons why **Development** is the most important phase in software life cycle.

(2 marks)

(c) Briefly explain the definition of the phase called **Maintenance** in this context.

(4 marks)

Question 2

(a) Define the term **composition** of object oriented design.

(2 marks)

(b) Given the following classes `Birthdate` and `Person`:

```
// CLASS BIRTHDATE
//------------------
class BirthDate
{
    private int day;
    private int month;
    private int year;

    public BirthDate(int d, int m, int y)
    {
        day = d;
        month = m;
        year = y;
    }

    public string GetDate()
```

```
        {
            return day+"/"+month+"/"+year;
        }

    }


    //CLASS PERSON
     //-------------
     class Person
     {
         private string name;
         private string ic;
         private BirthDate birthday;

         public Person(string n, string ic, BirthDate bd)
         {
             this.name = n;
             this.ic = ic;
             this.birthday = bd;
         }

         public void Print()
         {
            ...
         }
     }
```

i. Sketch the UML design to show the relationship of the classes `Birthdate` and `Person`.

(2 marks)

ii. Complete the function `Print()` in class `Person`.

(3 marks)

iii. Based on the answer in (ii), give the output of the following code:

```
Birthdate myBD = new Birthdate(12,1,1982);
Person thePerson = new Person(“Ahmad”,”820112-11-5121”,myBD);
thePerson.Print();
```

(3 marks)

Question 3

Given the code segment of a linked list object:

```
LinkedList<double> myLl = new LinkedList<double>();
myLl.AddFirst(12.9);
myLl.AddFirst(2.1);
myLl.AddLast(15.2);
```

(a) Based on the given code segment above, fill in the empty boxes in Figure 1:

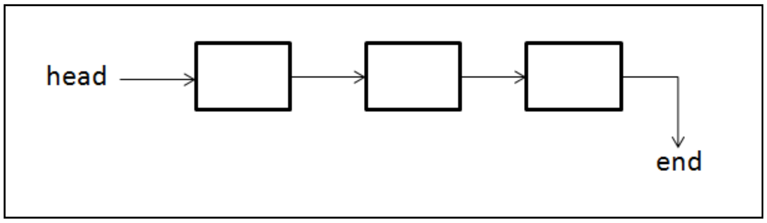


Figure 1 : linked list

(3 marks)

(b) Write the code segment to find the node that contains the value of **2.1** and put it in a new created `LinkedListNode` named `myNode`.

(1 mark)

(c) Write the code segment to insert a new node which contains the value of **3.3 after** `myNode`.

(1 mark)

(d) Create a `static void` function called `Display` which contains a parameter of type `LinkedList`. This function will print the values of the given parameter. Use the following header:

```
static void Display(LinkedList<double> theLL)
{

}
```

(3 marks)

(e) Based on the function created in (d), state the output of the following code segment.

```
Display(ll);
```

(2 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 (2) questions.

Please use the answer booklet provided.

Question 5

Given the following class:

```
class Rectangle
 {
     protected int length;
     protected int width;

     public Rectangle(int l, int w)
     {
         length = l;
         width = w;
     }

     public int GetArea()
     {
         return length * width;
     }

     public int GetCircumference()
     {
         return 2 * (length + width);
     }

     public void Print()
     {
         Console.WriteLine("The area of the rectangle is equal to " +
GetArea());
         Console.WriteLine("The circumferance of the rectangle is equal to
           " + GetCircumference());
     }
 }
```

(a) Give the output of the statements below by evaluating the object `r` of the class `Rectangle`.

```
Rectangle r = new Rectangle(10,15);
r.Print();
```

(5 marks)

(b) Given the following information:

"RectangularPrism inherits Rectangle"

According to the information above, write the header of the class `RectangularPrism`.

(1 mark)

(c) Write the constructor of the class RectangularPrism that contains **three (3) integer variables** which are used to initialize the length, width and height of a RectangularPrism.

(4 marks)

(d) Create a new method for the class RectangularPrism called GetVolume(). This method is used to calculate the volume of a rectangular prism.

(4 marks)

(e) RectangularPrism needs to overwrite the method GetArea() and Print() existing in Rectangle. Write the new headers of both methods in **base class** and **derived class**.

(4 marks)

(f) Write the definition of both overwritten methods in (e) using the following information:

- The method GetArea() in RectangularPrism is used to calculate the surface area of a rectangular prism which is equal to 2*length*width + 2*length*height + 2*width*height.
- The method Print() in RectangularPrism displays its area and its volume.

(5 marks)

(g) Give the output of the following code segment:

```
Rectangle r2 = new Rectangle(2,4);
RectangularPrism rp = new RectangularPrism(5,10,4);
r2.Print();
rp.Print();
```

(7 marks)

Question 6

As a software engineer, you are under a software development team in a company X. Your team is given a task to create a 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:

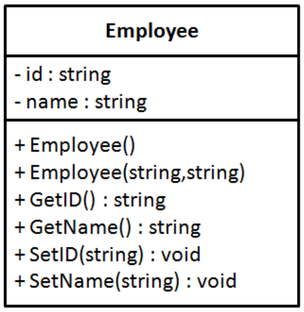
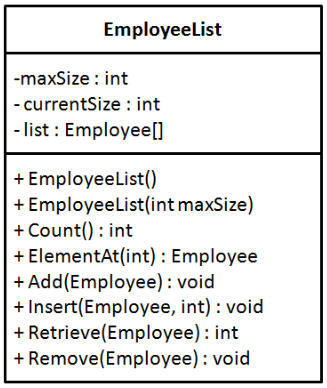


Figure 2 : UML Design of Employee



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 statement 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.

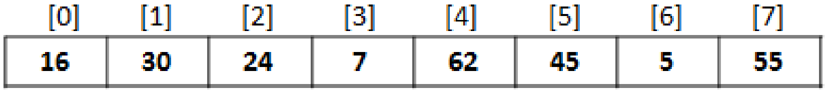
(10 marks)

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

(20 marks)

Question 7

Given the following list and program code:



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

Figure 4 : List of 8 integers

```
static void Main(string[] args)
{
    List<int> myList = new List<int>();
}

static int SequentialSearch(List<int> theList, int searchVal)
{
    for (int i = 0; i < theList.Count(); i++)
        if (theList.ElementAt(i) == searchVal)
            return i;
    return -1;
}

static List<int> InsertionSort(List<int> theList)
{
    int startInd = 1;
    int destInd;

    while (startInd < theList.Count())
    {
        destInd = -1;
        for (int i = 0; i < startInd - 1; i++)
        {
            if (theList.ElementAt(startInd) < theList.ElementAt(i))
            {
                destInd = i;
                break;
            }
        }

        if (destInd != -1)
        {
            int temp = theList.ElementAt(startInd);
            for (int i = startInd; i > destInd; i--)
            {
                theList[i] = theList[i - 1];
            }
            theList[destInd] = temp;
        }
        startInd++;

    }
    return theList;
}
```

(a) In `Main` function, add all the elements in the **figure 4** into the list `myList`.

(2 marks)

(b) Create a function with the following header to display all the elements in the given list of parameter.

```
static void Print(List<int> theList)
{

}
```

(3 marks)

(c) In `Main` function, add program code to perform the following statement:

- Declare an integer variable called `myVar` and prompt user to give a value.
- Search the existing of the given value from the list using `SequentialSearch` function.
- If the value exists, display its position. Otherwise, display that *"the value does not exist"*.

(5 marks)

(d) Create a function which checks whether the given list is sorted or unsorted. Use the following header:

```
static bool isSorted(List <int> theList)
{

}
```

(5 marks)

(e) Now, create a function to search a value from a list using **Binary Search** with the below requirements:

- If the list given is not sorted, the program needs to sort it first (use question (d)).
- The program consists of three indexes which are `startInd`, `lastInd` and `midInd`.
- `startInd` consists of the first index in search list while `lastInd` consists of the last index in search list. `midInd` is the middle point of `startInd` and `lastInd`.
- The program needs to compare between the search value and the value at `midInd`. If the search value is equal to the one at `midInd`, we will return `midInd` as the result and the program will stop. Otherwise, if the search value is greater than the one at `midInd`, `startInd` should be modified to `midInd+1`. But if it is smaller than the one at `midInd`, `lastInd` should be then modified to `midInd-1`.

This process will be repeated as long as `startInd` smaller or equal than `lastInd`.

- If the search value is not found in the list, the program should return -1 as the result.
- Use the following header:

```
static int BinarySearch(List <int> theList, int searchVal)
{

}
```

(10 marks)

(f) Elaborate the concept of **Selection Sort**.

(5 marks)

END OF QUESTION