C# and .NET Framework – Lab sheet 04

Classes and Objects

1. The class called Holiday is stated below.

An object of class Holiday represents a holiday during the year. This class has three instance variables:
name, which is a String representing the name of the holiday
day, which is an int representing the day of the month of the holiday
month, which is a String representing the month the holiday is in

```
public class Holiday {
        private String name;
        private int day;
        private String month;
        // your code goes here
}
```

a) Write a constructor for the class Holiday, which takes a String representing the name, an int representing the day, and a String representing the month as its arguments, and sets the class variables to these values.

```
public Holiday(string name, int day, string month)
{
    this.name = name;
    this.day = day;
    this.month = month;
}
```

b) Write a method inSameMonth, which compares two instances of the class Holiday, and returns the Boolean value true if they have the same month, and false if they do not.

```
public bool inSameMonth(Holiday other)
{
    return other.month == month;
}
```

c) Write a method avgDate which takes an array of base type Holiday as its argument, and returns a double that is the average of the day variables in the Holiday instances in the array. You may assume that the array is full (i.e. does not have any null entries).

```
public double avgDate(Holiday[] holidays)
{
    int dates = 0;
    foreach (var hDay in holidays)
    {
        dates += hDay.day;
    }
    return dates / (double)holidays.Length;
}
```

d) Write a piece of code that creates a Holiday instance with the name “Independence Day”, with the day “4”, and with the month “July”.

```
Holiday iDay = new Holiday("Independence Day", 4, "July");
```

2. The class Movie is stated below.

An instance of class Movie represents a film. This class has the following three class variables:
title, which is a String representing the title of the movie
studio, which is a String representing the studio that made the movie
rating, which is a String representing the rating of the movie (i.e. PG-13, R, etc)

```
public class Movie {
        private String title;
        private String studio;
        private String rating;
        // your code goes here
}
```

a) Write a constructor for the class Movie, which takes a String representing the title of the movie, a String representing the studio, and a String representing the rating as its arguments, and sets the respective class variables to these values.

```
public Movie(string title, string studio, string rating)
{
    this.title = title;
    this.studio = studio;
    this.rating = rating;
}
```

b) Write a second constructor for the class Movie, which takes a String representing the title of the movie and a String representing the studio as its arguments, and sets the respective class variables to these values, while the class variable rating is set to "PG".

```
public Movie(string title, string studio):this (title, studio, "PG") { }
```

c) Write a method getPG, which takes an array of base type Movie as its argument, and returns a new array of only those movies in the input array with a rating of "PG". You may assume the input array is full of Movie instances. The returned array need not be full.

```
public static Movie[] getPG(Movie[] movies)
{
     return movies.Where(m=>m.rating=="PG").ToArray();
}
```

d) Write a piece of code that creates an instance of the class Movie with the title “Casino Royale”, the studio “Eon Productions”, and the rating “PG13”

```
Movie cr = new Movie("Casino Royale", "Eon Productions", "PG13");
```

Method

Run this program, and explain the result.

```
using System;
namespace lab
{
    class basic_method
    {
        static void PrintLine()
        {
            for (int i = 0; i < 5; i++)
                Console.Write('*');
            Console.WriteLine();
        }
        static void Main()
        {
            PrintLine();
            PrintLine();
        }
    }
}
```

Output:

```
*****

*****
```

For loop is called 5 times, so 5 stars are printed and method is called twice so 2 times its printed.

a) Try to change the PrintLine( ) method to print 10 stars (*) in each line. Write your modified PrintLine() method

```
public static void PrintLine()
{
    for (int i = 0; i < 10; i++)
        Console.Write('*');
    Console.WriteLine();
}
```

b) From Example, try to change the Main() method to print 5 lines, where each line prints10 stars (*). Write your modified Main( ) method

```
public static void PrintLine()
{
    for (int i = 0; i < 10; i++)
        Console.Write('*');
    Console.WriteLine();
}
static void Main()
{
    for (int i = 0; i < 5; i++)
        PrintLine();
}
```

3. Parameter-passing method

```
static void PrintLine(char c, int k)
{
    for (int i = 0; i < k; i++)
        Console.Write(c);
    Console.WriteLine();
}
static void Main()
{
    int n;
    Console.Write("Input number of stars in each line:");
    n = int.Parse(Console.ReadLine());
    PrintLine('*', n);
}
```

a) Run this program, enter the values 3 and 7 for each run, write your results and explain them.

Input number of stars in each line:3

***

Input number of stars in each line:7

****

Loop in PrintLine runs for number of times passed as input.

b) Try to change the program to print lines of other characters (instead of "*"), and receive that character from the user. Write your modified Main() method?

```
static void Main()
{
    int n;
    Console.Write("Input number of chars in each line:");
    n = int.Parse(Console.ReadLine());
    Console.Write("Input the char to print: ");
    char c= Console.ReadKey().KeyChar;
    PrintLine(c, n);
}
```

```
Input number of chars in each line:5
Input the char to print: D
DDDDD
```

4. Return-value Method

```
using System;
namespace lab7
{
    class ex4
    {
        static int max(int a, int b)
        {
            // Input your code
        }
        static double avg(int a, int b)
        {
            // Input your code
        }
        static int mul(int a, int b)
        {
            // Input your code
        }
        static int celling(double a)
        {
            // Input your code
        }
        static void Main()
        {
            int i = int.Parse(Console.ReadLine());
            int j = int.Parse(Console.ReadLine());
            Console.WriteLine(" max = {0} ", max(i, j));
            Console.WriteLine(" avg = {0} ", avg(i, j));
            Console.WriteLine(" mul = {0} ", mul(i, j));
            Console.WriteLine(" celling = {0} ", celling(i));
        }
    }
}
```

a) Max

```
static int max(int a, int b)
{
    return Math.Max(a, b);
}
```

b) avg

```
static double avg(int a, int b)
{
    return (a + b) / 2;
}
```

c) mul

```
static int mul(int a, int b)
{
    return a * b;
}
```

d) celling

```
static int celling(double a)
{
    return Convert.ToInt32(Math.Ceiling(a));
}
```