

## Solutions for F1.PDF

### Q1) Rectangle and Tabletop (20 marks)

```
using System;

class Rectangle
{
    private double length;
    private double width;

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

    public double Area()
    {
        return length * width;
    }

    public void Display()
    {
        Console.WriteLine("Length = " + length);
        Console.WriteLine("Width = " + width);
        Console.WriteLine("Area = " + Area());
    }
}

class Tabletop : Rectangle
{
    public Tabletop(double l, double w) : base(l, w)
    {
    }

    public double Cost()
    {
        return Area() * 10;
    }

    public void DisplayAll()
    {
        Display();
        Console.WriteLine("Cost = " + Cost());
    }
}

class Program
{
    static void Main()
    {
        Tabletop t = new Tabletop(5, 3);
        t.DisplayAll();
    }
}
```

### Q2a) String Indexer (20 marks)

```
using System;

class MyIndexer
```

```

{
    private string[] arr = new string[3];

    public string this[int i]
    {
        get { return arr[i]; }
        set { arr[i] = value; }
    }
}

class Program
{
    static void Main()
    {
        MyIndexer m = new MyIndexer();
        m[0] = "C";
        m[1] = "C++";
        m[2] = "C#";

        Console.WriteLine(m[0]);
        Console.WriteLine(m[1]);
        Console.WriteLine(m[2]);
    }
}

```

### **Q2b) Dictionary (20 marks)**

```

using System;
using System.Collections.Generic;

class Program
{
    static void Main()
    {
        Dictionary<string, int> d = new Dictionary<string, int>();

        d.Add("Ali", 80);
        d.Add("Ahmed", 90);
        d.Add("Sara", 70);

        if (d.ContainsKey("Ali"))
        {
            d["Ali"] = 95;
        }

        foreach (var x in d)
        {
            Console.WriteLine(x.Key + " : " + x.Value);
        }
    }
}

```

### **Q3) Short Instructions (20 marks)**

- 1) private int Calculation(int a, int b)
 {
 return a + b;
 }
- 2) float[] Degree = { 2.5f, 3f, 8.5f };
- 3) Array.Sort(Nums);
 Array.Reverse(Nums);

```

4) public override void AnimalSound()
{
    Console.WriteLine("Sound");
}

5) class DataStore<T>
{
    public T Data { get; set; }
}

6) Dictionary<string, string> Cities = new Dictionary<string, string>()
{
    { "Baghdad", "Iraq" }
};

7) arList.RemoveRange(0, 2);

```

**Q4a) Float Array \* Integer (20 marks)**

```

using System;

class MyClass
{
    private float[] arr = new float[3];
    private int num;

    public MyClass()
    {
        Console.Write("Enter arr[0]: ");
        arr[0] = float.Parse(Console.ReadLine());

        Console.Write("Enter arr[1]: ");
        arr[1] = float.Parse(Console.ReadLine());

        Console.Write("Enter arr[2]: ");
        arr[2] = float.Parse(Console.ReadLine());

        Console.Write("Enter num: ");
        num = int.Parse(Console.ReadLine());
    }

    public void Multiply()
    {
        arr[0] = arr[0] * num;
        arr[1] = arr[1] * num;
        arr[2] = arr[2] * num;
    }

    public void Display()
    {
        Console.WriteLine(arr[0]);
        Console.WriteLine(arr[1]);
        Console.WriteLine(arr[2]);
    }
}

class Program
{
    static void Main()
    {
        MyClass m = new MyClass();
        m.Multiply();
        m.Display();
    }
}

```

#### **Q4b) Count Letter a (20 marks)**

```
using System;

class MyClass
{
    private string[] names = new string[3];
    private int[] counts = new int[3];

    public MyClass()
    {
        Console.Write("Enter name[0]: ");
        names[0] = Console.ReadLine();

        Console.Write("Enter name[1]: ");
        names[1] = Console.ReadLine();

        Console.Write("Enter name[2]: ");
        names[2] = Console.ReadLine();
    }

    public void CountA()
    {
        for (int i = 0; i < 3; i++)
        {
            counts[i] = 0;
            foreach (char c in names[i])
            {
                if (c == 'a' || c == 'A')
                    counts[i]++;
            }
        }
    }

    public void Display()
    {
        Console.WriteLine(names[0] + " has " + counts[0] + " a");
        Console.WriteLine(names[1] + " has " + counts[1] + " a");
        Console.WriteLine(names[2] + " has " + counts[2] + " a");
    }
}

class Program
{
    static void Main()
    {
        MyClass m = new MyClass();
        m.CountA();
        m.Display();
    }
}
```

#### **Q5) Short Instructions (20 marks)**

- 1) 

```
Console.WriteLine("Enter Prime: ");
int Prime = int.Parse(Console.ReadLine());
```
- 2) 

```
double Multiply(params float[] nums)
{
    double result = 1;
    foreach (float n in nums)
        result = result * n;
    return result;
}
```

```
3) foreach (var item in Thing)
{
    Console.WriteLine(item);
}

4) public string Stage
{
    set { }
}

5) enum Size
{
    Small = 0,
    Med = 5,
    Large = 6
}

6) List<string> Country = new List<string>();

7) float f;
bool ok = float.TryParse(Console.ReadLine(), out f);
if (!ok)
    Console.WriteLine("Wrong input");
```

## Solutions for F2.PDF

### Q1) Vehicle and Car (20 marks)

```
using System;

class Vehicle
{
    protected string company;
    protected string country;
    protected int year;

    public Vehicle()
    {
        Console.Write("Enter company: ");
        company = Console.ReadLine();

        Console.Write("Enter country: ");
        country = Console.ReadLine();

        Console.Write("Enter year: ");
        year = int.Parse(Console.ReadLine());
    }

    public virtual void Display()
    {
        Console.WriteLine("Company: " + company);
        Console.WriteLine("Country: " + country);
        Console.WriteLine("Year: " + year);
    }
}

class Car : Vehicle
{
    private string model;

    public Car()
    {
        Console.Write("Enter model: ");
        model = Console.ReadLine();
    }

    public override void Display()
    {
        Console.WriteLine("Model: " + model);
    }

    public void Honk()
    {
        Console.WriteLine("Beep Beep!");
    }
}

class Program
{
    static void Main()
    {
        Vehicle v = new Vehicle();
        v.Display();

        Vehicle c = new Car();
        c.Display();
    }
}
```

**Q2a) List (20 marks)**

```
using System;
using System.Collections.Generic;

class Program
{
    static void Main()
    {
        List<int> nums = new List<int>();

        nums.Add(10);
        nums.Add(20);
        nums.Add(30);
        nums.Add(40);
        nums.Add(50);

        nums.RemoveAt(1);

        if (nums.Contains(30))
        {
            Console.WriteLine("30 exists");
        }

        foreach (int n in nums)
        {
            Console.WriteLine(n);
        }
    }
}
```

**Q2b) ArrayList (20 marks)**

```
using System;
using System.Collections;

class Program
{
    static void Main()
    {
        ArrayList al = new ArrayList();

        al.Add(10);
        al.Add("Hello");
        al.Add(3.14);
        al.Add(true);

        Console.WriteLine(al[1]);

        al.Remove(3.14);

        foreach (var item in al)
        {
            Console.WriteLine(item);
        }
    }
}
```

**Q3) Short Instructions (20 marks)**

1) public static int Play()  
{

```

        return 0;
    }

2) int[] Age = { 1, 2, 3 };

3) Array.Sort(Nums);

4) public virtual void AnimalSound()
{
}

5) class Printer
{
    public void Print<T>(T item)
    {
        Console.WriteLine(item);
    }
}

6) Dictionary<int, string> numberNames = new Dictionary<int, string>()
{
    { 1, "One" }
};

7) bool exists = List1.Contains(5);

```

#### **Q4a) Count 1 in Float (20 marks)**

```

using System;

class MyClass
{
    private float[] fArr = new float[3];
    private int[] iArr = new int[3];

    public MyClass()
    {
        Console.Write("Enter f[0]: ");
        fArr[0] = float.Parse(Console.ReadLine());

        Console.Write("Enter f[1]: ");
        fArr[1] = float.Parse(Console.ReadLine());

        Console.Write("Enter f[2]: ");
        fArr[2] = float.Parse(Console.ReadLine());
    }

    public void Count1()
    {
        for (int i = 0; i < 3; i++)
        {
            iArr[i] = 0;
            string s = fArr[i].ToString();
            foreach (char c in s)
            {
                if (c == '1')
                    iArr[i]++;
            }
        }
    }

    public void Display()
    {
        Console.WriteLine(fArr[0] + " has " + iArr[0] + " ones");
        Console.WriteLine(fArr[1] + " has " + iArr[1] + " ones");
        Console.WriteLine(fArr[2] + " has " + iArr[2] + " ones");
    }
}

```

```

    }
}

class Program
{
    static void Main()
    {
        MyClass m = new MyClass();
        m.Count1();
        m.Display();
    }
}

```

#### **Q4b) Count Letter b (20 marks)**

```

using System;

class MyClass
{
    public string[] names = new string[3];
    public int[] counts = new int[3];

    public MyClass()
    {
        Console.Write("Enter name[0]: ");
        names[0] = Console.ReadLine();

        Console.Write("Enter name[1]: ");
        names[1] = Console.ReadLine();

        Console.Write("Enter name[2]: ");
        names[2] = Console.ReadLine();
    }

    public void CountB()
    {
        for (int i = 0; i < 3; i++)
        {
            counts[i] = 0;
            foreach (char c in names[i])
            {
                if (c == 'b' || c == 'B')
                    counts[i]++;
            }
        }
    }

    public void Display()
    {
        Console.WriteLine(names[0] + " has " + counts[0] + " b");
        Console.WriteLine(names[1] + " has " + counts[1] + " b");
        Console.WriteLine(names[2] + " has " + counts[2] + " b");
    }
}

class Program
{
    static void Main()
    {
        MyClass m = new MyClass();
        m.CountB();
        m.Display();
    }
}

```

#### **Q5) Short Instructions (20 marks)**

```
1) Console.WriteLine("Enter Price: ");
   float Price = float.Parse(Console.ReadLine());

2) long Sum(params int[] nums)
{
    long result = 0;
    foreach (int n in nums)
        result = result + n;
    return result;
}

3) foreach (var item in Collections)
{
    Console.WriteLine(item);
}

4) public string Name { get; set; }

5) enum Level
{
    Low = 2,
    Med = 3,
    High = 4
}

6) ArrayList Info = new ArrayList() { 4, 'A', 3.5, true, "Bill" };

7) int i;
   bool ok = int.TryParse(Console.ReadLine(), out i);
   if (!ok)
       Console.WriteLine("Wrong input");
```