

Solutions for F1.PDF

Q1) Rectangle and Tabletop Classes (20 marks)

Create base class Rectangle with constructor, area calculation and display methods. Derived class Tabletop calculates cost and displays all values.

```
using System;
class Rectangle
{
    private double length, width;
    public Rectangle(double l, double w) { length = l; width = w; }
    public double CalculateArea() { return length * width; }
    public void Display() { Console.WriteLine($"Length: {length}, Width: {width}, Area: {CalculateArea()}"); }
    public double Length { get { return length; } }
    public double Width { get { return width; } }
}
class Tabletop : Rectangle
{
    private double costPerUnit = 10;
    public Tabletop(double l, double w) : base(l, w) { }
    public double CalculateCost() { return CalculateArea() * costPerUnit; }
    public void DisplayAll() { Display(); Console.WriteLine($"Cost: {CalculateCost()}"); }
}
class Program
{
    static void Main()
    {
        Tabletop t = new Tabletop(5, 3);
        t.DisplayAll();
    }
}
```

Q2a) String Indexer (20 marks)

Create string indexer that stores three programming language names.

```
using System;
class StringIndexer
{
    private string[] languages = new string[3];
    public string this[int index]
    {
        get { return languages[index]; }
        set { languages[index] = value; }
    }
}
class Program
{
    static void Main()
    {
        StringIndexer s = new StringIndexer();
        s[0] = "C";
        s[1] = "C++";
        s[2] = "C#";
        for (int i = 0; i < 3; i++) Console.WriteLine(s[i]);
    }
}
```

Q2b) Dictionary with Student Scores (20 marks)

Create dictionary with string keys and integer values for student scores.

```
using System;
using System.Collections.Generic;
class Program
```

```

{
    static void Main()
    {
        Dictionary<string, int> scores = new Dictionary<string, int>();
        scores.Add("Ali", 85);
        scores.Add("Ahmed", 90);
        scores.Add("Sara", 78);
        if (scores.ContainsKey("Ali")) scores["Ali"] = 95;
        foreach (var item in scores)
            if (item.Value >= 50) Console.WriteLine($"{item.Key}: {item.Value}");
    }
}

```

Q3) Write Instructions (20 marks)

Short code snippets for various C# concepts.

```

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) Class with Float Array and Integer (20 marks)

Class with float array and integer, multiply array elements by the integer.

```

using System;
class MyClass
{
    private float[] arr = new float[3];
    private int num;
    public MyClass()
    {
        for (int i = 0; i < 3; i++) { Console.Write($"arr[{i}]: "); arr[i] = float.Parse(Console.ReadLine()); }
        Console.Write("num: "); num = int.Parse(Console.ReadLine());
    }
    public void Multiply() { for (int i = 0; i < 3; i++) arr[i] *= num; }
    public void Display() { foreach (float f in arr) Console.WriteLine(f); }
}
class Program
{
    static void Main() { MyClass m = new MyClass(); m.Multiply(); m.Display(); }
}

```

Q4b) Count Letter 'a' in Strings (20 marks)

Class with string array and integer array, count letter 'a' in each string.

```

using System;
class MyClass
{
    private string[] names = new string[3];
    private int[] counts = new int[3];
    public MyClass()
    {
        for (int i = 0; i < 3; i++) { Console.Write($"name[{i}]: "); names[i] = Console.ReadLine(); }
    }
    public void CountA()
    {

```

```

        for (int i = 0; i < 3; i++)
            foreach (char c in names[i]) if (c == 'a' || c == 'A') counts[i]++;
    }
    public void Display() { for (int i = 0; i < 3; i++) Console.WriteLine($"{names[i]}: {counts[i]}"); }
}
class Program
{
    static void Main() { MyClass m = new MyClass(); m.CountA(); m.Display(); }
}

```

Q5) Write Instructions (20 marks)

Short code snippets for various C# concepts.

1. `Console.Write("Prime: "); int Prime = int.Parse(Console.ReadLine());`
2. `double Multiply(params float[] nums) { double r = 1; foreach(var n in nums) r *= n; return r; }`
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; while (!float.TryParse(Console.ReadLine(), out f)) Console.Write("Invalid: ");`

Solutions for F2.PDF

Q1) Vehicle and Car Classes (20 marks)

Base class Vehicle with protected variables and derived class Car with model and honk method.

```
using System;
class Vehicle
{
    protected string company, country;
    protected int year;
    public Vehicle()
    {
        Console.WriteLine("Company: "); company = Console.ReadLine();
        Console.WriteLine("Country: "); country = Console.ReadLine();
        Console.WriteLine("Year: "); year = int.Parse(Console.ReadLine());
    }
    public virtual void Display() { Console.WriteLine($"{company}, {country}, {year}"); }
}
class Car : Vehicle
{
    private string model;
    public Car()
    {
        Console.WriteLine("Model: "); model = Console.ReadLine();
    }
    public override void Display() { Console.WriteLine($"Model: {model}"); }
    public void Honk() { Console.WriteLine("Beep!"); }
}
class Program
{
    static void Main()
    {
        Vehicle v = new Vehicle();
        v.Display();
        Vehicle c = new Car();
        c.Display();
    }
}
```

Q2a) List of Integers (20 marks)

Create List with 5 integers, remove by index, check existence, print all.

```
using System;
using System.Collections.Generic;
class Program
{
    static void Main()
    {
        List<int> nums = new List<int> { 1, 2, 3, 4, 5 };
        nums.RemoveAt(1);
        if (nums.Contains(3)) Console.WriteLine("3 exists");
        foreach (int n in nums) Console.WriteLine(n);
    }
}
```

Q2b) ArrayList with Different Types (20 marks)

Create ArrayList with different types, display by index, remove by value.

```
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) Write Instructions (20 marks)

Short code snippets for various C# concepts.

```

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 Number 1 in Float (20 marks)

Class with float array, count digit 1 in each float and store in integer array.

```

using System;
class MyClass
{
    private float[] fArr = new float[3];
    private int[] iArr = new int[3];
    public MyClass()
    {
        for (int i = 0; i < 3; i++) { Console.Write($"f[{i}]: "); fArr[i] = float.Parse(Console.ReadLine()); }
    }
    public void Count1()
    {
        for (int i = 0; i < 3; i++)
            foreach (char c in fArr[i].ToString()) if (c == '1') iArr[i]++;
    }
    public void Display() { for (int i = 0; i < 3; i++) Console.WriteLine($"{fArr[i]}: {iArr[i]}"); }
}
class Program
{
    static void Main() { MyClass m = new MyClass(); m.Count1(); m.Display(); }
}

```

Q4b) Count Letter 'b' in Strings (20 marks)

Class with string array and integer array, count letter 'b' in each string.

```

using System;
class MyClass
{
    public string[] names = new string[3];
    public int[] counts = new int[3];
    public MyClass()
    {
        for (int i = 0; i < 3; i++) { Console.Write($"name[{i}]: "); names[i] = Console.ReadLine(); }
    }
}

```

```

    public void CountB()
    {
        for (int i = 0; i < 3; i++)
            foreach (char c in names[i]) if (c == 'b' || c == 'B') counts[i]++;
    }
    public void Display() { for (int i = 0; i < 3; i++) Console.WriteLine($"{names[i]}: {counts[i]}"); }
}
class Program
{
    static void Main() { MyClass m = new MyClass(); m.CountB(); m.Display(); }
}

```

Q5) Write Instructions (20 marks)

Short code snippets for various C# concepts.

```

1. Console.Write("Price: "); float Price = float.Parse(Console.ReadLine());

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

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; while (!int.TryParse(Console.ReadLine(), out i)) Console.Write("Invalid: ");

```