

C# Programs With Outputs

PART-A

1. Convert Celsius to Fahrenheit

using System;

```
class Program {  
    static void Main() {  
        Console.Write("Enter temperature in Celsius: ");  
        double c = Convert.ToDouble(Console.ReadLine());  
        double f = (c * 9 / 5) + 32;  
        Console.WriteLine($"Temperature in Fahrenheit: {f}");  
    }  
}
```

Output:

Enter temperature in Celsius: 37

Temperature in Fahrenheit: 98.6

2. Check whether a number is even or odd

using System;

```
class Program {  
    static void Main() {  
        Console.Write("Enter a number: ");  
        int n = Convert.ToInt32(Console.ReadLine());  
        if (n % 2 == 0)  
            Console.WriteLine($"{n} is Even");  
        else  
            Console.WriteLine($"{n} is Odd");  
    }  
}
```

Output:

Enter a number: 7

7 is Odd

3. Sum of multiples of 3 and 5

using System;

```
class Program {  
    static void Main() {  
        int sum = 0;  
        for (int i = 1; i <= 100; i++) {  
            if (i % 3 == 0 || i % 5 == 0)  
                sum += i;  
        }  
        Console.WriteLine("Sum of multiples of 3 and 5 from 1 to 100: " + sum);  
    }  
}
```

Output:

Sum of multiples of 3 and 5 from 1 to 100: 2418

4. Find minimum and maximum from given data

using System;

```
class Program {  
    static void Main() {  
        int[] arr = { 12, 45, 7, 89, 34 };  
        int min = arr[0], max = arr[0];  
        foreach (int i in arr) {  
            if (i < min) min = i;  
            if (i > max) max = i;  
        }  
        Console.WriteLine($"Minimum = {min}, Maximum = {max}");  
    }  
}
```

Output:

Minimum = 7, Maximum = 89

5. Demonstrate conditional logical operator

using System;

```
class Program {  
    static void Main() {  
        int a = 10, b = 20;  
        if (a < b && b == 20)  
            Console.WriteLine("Both conditions are true");  
        else  
            Console.WriteLine("Conditions are false");  
    }  
}
```

Output:

Both conditions are true

6. Generate register numbers using static constructor

using System;

```
class Student {  
    static int count;  
    int regNo;  
    static Student() {  
        count = 1000;  
    }  
    public Student() {  
        regNo = ++count;  
    }  
    public void Display() {  
        Console.WriteLine("Register Number: " + regNo);  
    }  
}
```

```

    }
}
class Program {
    static void Main() {
        for (int i = 0; i < 10; i++) {
            Student s = new Student();
            s.Display();
        }
    }
}

```

Output:

Register Number: 1001
 Register Number: 1002
 ...
 Register Number: 1010

7. Generate marksheet of students

```

using System;
class Program {
    static void Main() {
        Console.Write("Enter number of subjects: ");
        int n = int.Parse(Console.ReadLine());
        int[] marks = new int[n];
        int total = 0;

        for (int i = 0; i < n; i++) {
            Console.Write($"Enter marks for subject {i + 1}: ");
            marks[i] = int.Parse(Console.ReadLine());
            total += marks[i];
        }

        double avg = total / (double)n;
        Console.WriteLine($"Total = {total}, Average = {avg}");

        if (avg >= 50)
            Console.WriteLine("Result: PASS");
        else
            Console.WriteLine("Result: FAIL");
    }
}

```

Output:

Enter number of subjects: 3
 Enter marks for subject 1: 60
 Enter marks for subject 2: 75
 Enter marks for subject 3: 80
 Total = 215, Average = 71.67
 Result: PASS

8. Check if a given year is leap year

using System;

```
class Program {  
    static void Main() {  
        Console.Write("Enter a year: ");  
        int year = int.Parse(Console.ReadLine());  
        if ((year % 400 == 0) || (year % 4 == 0 && year % 100 != 0))  
            Console.WriteLine($"{year} is a Leap Year");  
        else  
            Console.WriteLine($"{year} is not a Leap Year");  
    }  
}
```

Output:

Enter a year: 2024

2024 is a Leap Year

9. Search an element in an array

using System;

```
class Program {  
    static void Main() {  
        int[] arr = { 10, 20, 30, 40, 50 };  
        Console.Write("Enter element to search: ");  
        int key = int.Parse(Console.ReadLine());  
        bool found = false;  
        foreach (int i in arr) {  
            if (i == key) {  
                found = true;  
                break;  
            }  
        }  
        if (found)  
            Console.WriteLine($"{key} found in array");  
        else  
            Console.WriteLine($"{key} not found in array");  
    }  
}
```

Output:

Enter element to search: 30

30 found in array

10. Print lower and upper bound of array

using System;

```
class Program {  
    static void Main() {  
        int[] arr = { 5, 10, 15, 20 };  
        Console.WriteLine("Lower Bound: " + arr.GetLowerBound(0));  
    }  
}
```

```
        Console.WriteLine("Upper Bound: " + arr.GetUpperBound(0));
    }
}
```

Output:

Lower Bound: 0

Upper Bound: 3

11. Check whether number is Armstrong or not

using System;

```
class Program {
    static void Main() {
        Console.Write("Enter a number: ");
        int num = int.Parse(Console.ReadLine());
        int temp = num, sum = 0, digit;
        while (temp > 0) {
            digit = temp % 10;
            sum += digit * digit * digit;
            temp /= 10;
        }
        if (sum == num)
            Console.WriteLine($"{num} is an Armstrong number");
        else
            Console.WriteLine($"{num} is not an Armstrong number");
    }
}
```

Output:

Enter a number: 153

153 is an Armstrong number

C# PROGRAMS WITH OUTPUTS

PART-B

1. Convert character of a string to opposite case

using System;

```
class Program {
    static void Main() {
        Console.Write("Enter a string: ");
        string str = Console.ReadLine();
        string result = "";
        foreach (char c in str) {
            if (char.IsUpper(c))
                result += char.ToLower(c);
            else if (char.IsLower(c))
                result += char.ToUpper(c);
        }
    }
}
```

```

        else
            result += c;
    }
    Console.WriteLine("Converted string: " + result);
}
}

```

Output:

Enter a string: HeLLo WoRLd
 Converted string: hElLO wOrID

2. Demonstrate class and object creation

```

using System;
class Student {
    public int id;
    public string name;

    public void GetDetails() {
        Console.Write("Enter ID: ");
        id = int.Parse(Console.ReadLine());
        Console.Write("Enter Name: ");
        name = Console.ReadLine();
    }

    public void Display() {
        Console.WriteLine($"ID: {id}, Name: {name}");
    }
}

class Program {
    static void Main() {
        Student s1 = new Student();
        s1.GetDetails();
        s1.Display();
    }
}

```

Output:

Enter ID: 1
 Enter Name: Rahul
 ID: 1, Name: Rahul

3. Demonstrate “this” reference with example

```

using System;
class Student {
    int id;
    string name;

```

```

    public Student(int id, string name) {
        this.id = id;
        this.name = name;
    }

    public void Display() {
        Console.WriteLine($"ID: {id}, Name: {name}");
    }
}

class Program {
    static void Main() {
        Student s = new Student(101, "Anita");
        s.Display();
    }
}

```

Output:

ID: 101, Name: Anita

4. Demonstrate Multilevel Inheritance

```

using System;

class Person {
    public void PersonInfo() {
        Console.WriteLine("This is Person class");
    }
}

class Student : Person {
    public void StudentInfo() {
        Console.WriteLine("This is Student class");
    }
}

class Graduate : Student {
    public void GraduateInfo() {
        Console.WriteLine("This is Graduate class");
    }
}

class Program {
    static void Main() {
        Graduate g = new Graduate();
        g.PersonInfo();
        g.StudentInfo();
        g.GraduateInfo();
    }
}

```

Output:

This is Person class

This is Student class

This is Graduate class

5. Demonstrate Multiple Exceptions

using System;

```
class Program {  
    static void Main() {  
        try {  
            int[] arr = new int[3];  
            arr[5] = 10; // Index out of range  
            int x = 10 / 0; // Divide by zero  
        }  
        catch (DivideByZeroException e) {  
            Console.WriteLine("Error: " + e.Message);  
        }  
        catch (IndexOutOfRangeException e) {  
            Console.WriteLine("Error: " + e.Message);  
        }  
        finally {  
            Console.WriteLine("Finally block executed.");  
        }  
    }  
}
```

Output:

Error: Index was outside the bounds of the array.

Finally block executed.

6. Demonstrate NullReferenceException

using System;

```
class Program {  
    static void Main() {  
        try {  
            string s = null;  
            Console.WriteLine(s.Length);  
        }  
        catch (NullReferenceException e) {  
            Console.WriteLine("Caught Exception: " + e.Message);  
        }  
    }  
}
```

Output:

Caught Exception: Object reference not set to an instance of an object.

7. Create a Simple Thread

using System;

using System.Threading;

```
class Program {
```



```

static void Display() {
    for (int i = 1; i <= 5; i++) {
        Console.WriteLine("Thread running... " + i);
        Thread.Sleep(500);
    }
}

static void Main() {
    Thread t = new Thread(Display);
    t.Start();
}
}

```

Output:

```

Thread running... 1
Thread running... 2
Thread running... 3
Thread running... 4
Thread running... 5

```

8. Create Thread Pools

```

using System;
using System.Threading;

class Program {
    static void Task(Object obj) {
        Console.WriteLine("Task executed by thread: " +
Thread.CurrentThread.ManagedThreadId);
    }

    static void Main() {
        for (int i = 0; i < 5; i++) {
            ThreadPool.QueueUserWorkItem(Task);
        }
        Thread.Sleep(1000); // wait for threads to complete
    }
}

```

Output:

```

Task executed by thread: 4
Task executed by thread: 5
Task executed by thread: 3
Task executed by thread: 6
Task executed by thread: 7

```

9. Design a simple Login Form (Console-based)

```

using System;
class Program {

```

```

static void Main() {
    string username = "admin";
    string password = "1234";

    Console.Write("Enter username: ");
    string u = Console.ReadLine();
    Console.Write("Enter password: ");
    string p = Console.ReadLine();

    if (u == username && p == password)
        Console.WriteLine("Login Successful!");
    else
        Console.WriteLine("Invalid username or password.");
}
}

```

Output:

```

Enter username: admin
Enter password: 1234
Login Successful!

```

10. Design a To-Do List (basic version)

```

using System;
using System.Collections.Generic;

class Program {
    static void Main() {
        List<string> todo = new List<string>();
        while (true) {
            Console.WriteLine("\n1. Add Task 2. View Tasks 3. Exit");
            Console.Write("Enter choice: ");
            int choice = int.Parse(Console.ReadLine());

            if (choice == 1) {
                Console.Write("Enter task: ");
                string task = Console.ReadLine();
                todo.Add(task);
                Console.WriteLine("Task added!");
            }
            else if (choice == 2) {
                Console.WriteLine("Your To-Do List:");
                foreach (string t in todo)
                    Console.WriteLine("- " + t);
            }
            else if (choice == 3) {
                break;
            }
            else {

```

```

        Console.WriteLine("Invalid choice.");
    }
}
}
}

```

Output (example run):

1. Add Task 2. View Tasks 3. Exit
Enter choice: 1
Enter task: Complete C# assignment
Task added!

1. Add Task 2. View Tasks 3. Exit
Enter choice: 2
Your To-Do List:
- Complete C# assignment

11. Design a Simple Calculator

```

using System;
class Program {
    static void Main() {
        Console.Write("Enter first number: ");
        double a = Convert.ToDouble(Console.ReadLine());
        Console.Write("Enter operator (+, -, *, /): ");
        char op = Convert.ToChar(Console.ReadLine());
        Console.Write("Enter second number: ");
        double b = Convert.ToDouble(Console.ReadLine());
        double result = 0;

        switch (op) {
            case '+': result = a + b; break;
            case '-': result = a - b; break;
            case '*': result = a * b; break;
            case '/': result = b != 0 ? a / b : double.NaN; break;
            default: Console.WriteLine("Invalid operator"); return;
        }
        Console.WriteLine($"Result: {result}");
    }
}

```

Output:

Enter first number: 10
Enter operator (+, -, *, /): *
Enter second number: 5
Result: 50
