

# OOP with C#

## Lab Sheet (part 2)

04. using System;

class Program

{

static void Main()

{

// user to enter the value of N

Console.Write("Enter the value of N for Fibonacci series: ");

int n = Convert.ToInt32(Console.ReadLine());

// Fibonacci series using recursion

Console.WriteLine("\nFibonacci Series:");

for (int i = 0; i < n; i++)

{

Console.Write(Fibonacci(i) + " ");

}

// Wait for user input before closing the console window

Console.ReadLine();

}

// Recursive function to calculate the Fibonacci series

```

static int Fibonacci(int n)
{
    if (n <= 1)
    {
        return n;
    }
    else
    {
        return Fibonacci(n - 1) + Fibonacci(n - 2);
    }
}
}

```

**05.** using System;

```

class Program
{
    static void Main()
    {
        // user to enter a number
        Console.Write("Enter a number to display its multiplication table: ");
        int number = Convert.ToInt32(Console.ReadLine());

        // multiplication table using a loop
        Console.WriteLine($"
Multiplication Table of {number}:");

        for (int i = 1; i <= 10; i++)

```

```

    {
        Console.WriteLine($"{number} x {i} = {number * i}");
    }

    // Wait for user input before closing the console window
    Console.ReadLine();
}
}

```

**06.** using System;

class Program

```

{
    static void Main()
    {
        // user to input a student's name
        Console.Write("Enter student's name: ");
        string studentName = Console.ReadLine();

        // user to input exam marks with validation
        int examMarks;
        do
        {
            Console.Write("Enter exam marks (0-100): ");
            if (!int.TryParse(Console.ReadLine(), out examMarks) || examMarks < 0 || examMarks >
100)
            {

```

```

        Console.WriteLine("Error: Please enter a valid mark between 0 and 100.");
    }
} while (examMarks < 0 || examMarks > 100);

// Determine and display the corresponding grade
string grade = CalculateGrade(examMarks);

// student's name along with their assigned grade
Console.WriteLine($"{Student Name: {studentName}}");
Console.WriteLine($"Grade: {grade}");

// Wait for user input before closing the console window
Console.ReadLine();
}

// Function to calculate the grade based on exam marks
static string CalculateGrade(int marks)
{
    if (marks >= 75 && marks <= 100)
    {
        return "A";
    }
    else if (marks >= 60 && marks <= 74)
    {
        return "B";
    }
    else if (marks >= 50 && marks <= 59)

```

```

    {
        return "C";
    }
    else if (marks >= 40 && marks <= 49)
    {
        return "D";
    }
    else
    {
        return "Fail";
    }
}
}

```

**07.** using System;

class Program

```

{
    static double accountBalance = 1000; // Initial account balance

    static void Main()
    {
        DisplayMenu();
    }

    static void DisplayMenu()
    {

```

```
while (true)
{
    Console.WriteLine("\nATM Menu:");
    Console.WriteLine("1. Check Balance");
    Console.WriteLine("2. Deposit Money");
    Console.WriteLine("3. Withdraw Money");
    Console.WriteLine("4. Exit");
    Console.Write("Enter your choice (1-4): ");

    int choice;
    if (int.TryParse(Console.ReadLine(), out choice))
    {
        switch (choice)
        {
            case 1:
                CheckBalance();
                break;
            case 2:
                DepositMoney();
                break;
            case 3:
                WithdrawMoney();
                break;
            case 4:
                Console.WriteLine("Exiting ATM. Have a nice day!");
                return;
            default:
```

```
        Console.WriteLine("Invalid choice. Please enter a number between 1 and 4.");
        break;
    }
}
else
{
    Console.WriteLine("Invalid input. Please enter a valid number.");
}
}
}
```

```
static void CheckBalance()
```

```
{
    Console.WriteLine($"Your account balance is: ${accountBalance}");
}
```

```
static void DepositMoney()
```

```
{
    Console.Write("Enter the amount to deposit: $");
    double amount;
    if (double.TryParse(Console.ReadLine(), out amount) && amount > 0)
    {
        accountBalance += amount;
        Console.WriteLine($"${amount} deposited successfully.");
        Console.WriteLine($"Your updated account balance is: ${accountBalance}");
    }
    else
```

```
{  
    Console.WriteLine("Invalid amount. Please enter a valid positive number.");  
}  
}
```

```
static void WithdrawMoney()
```

```
{  
    Console.Write("Enter the amount to withdraw: $");  
    double amount;  
    if (double.TryParse(Console.ReadLine(), out amount) && amount > 0)  
    {  
        if (amount <= accountBalance)  
        {  
            accountBalance -= amount;  
            Console.WriteLine($"{amount} withdrawn successfully.");  
            Console.WriteLine($"Your updated account balance is: ${accountBalance}");  
        }  
        else  
        {  
            Console.WriteLine("Insufficient funds. Withdrawal failed.");  
        }  
    }  
    else  
    {  
        Console.WriteLine("Invalid amount. Please enter a valid positive number.");  
    }  
}}
```



