Mid Exam - Questions 2 & 3 Solutions

████ ██████ ██████ ███████
██ █████ ███████

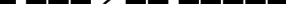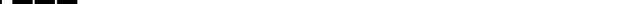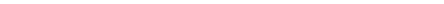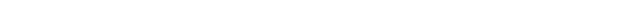## Question 2 (40 Marks)

███████ ███████ (40 ██████)
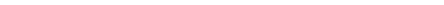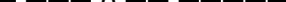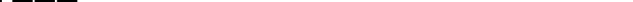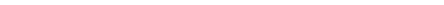
Question: Write a complete C# program which contains one dimensional array of 10 integer numbers. Let the user insert the numbers through the keyboard. After entering the numbers let the program count the occurrence of odd and even numbers in the array and display the result on the screen by using foreach loop.

██████: ████ ███████ C# ████ ██████ ███ ███████ ██████ █████ ██ 10 █████ █████. ██ ████████ █████ ██████ ██ ████ ████ ████████. ███ █████ ████████ ██ ████████ ███ █████ ████████ ███████ ███████ ██ ████████ █████ ███████ ███ ██████ ████████ ████ foreach.

Solution Explanation / ███ ████:

Step-by-step approach (███████):
1. Create an integer array of size 10 (█████ ██████ ██ 10 █████)
2. Use a for loop to get input from user (████████ ████ for ██████ ███████)
3. Initialize counters for odd and even numbers (█████ ██████ ███████ ███████ ████████)
4. Use foreach loop to iterate through array (████████ ████ foreach ██████ ███ ████████)
5. Check if number is odd or even using modulo operator (%) (███ ███ ███ █████ █████ ██ █████ ████████ %)
6. Display the results (███ ████████)

Key Concepts (████████ ████████):
• A number is **even** if (number % 2 == 0) - █████ ████ ███ ███ ██████ ██ ██████ ███ 2 █████ ███
• A number is **odd** if (number % 2 != 0) - █████ ████ ███ ███ ██████ ██ ██████ ███ 2 ██ █████ ███
• **foreach** loop is used to iterate through collections - ████ foreach ██████ ██████

Complete C# Program:

```csharp
using System;

class Program
{
    static void Main()
    {
        // Step 1: Declare an array of 10 integers
        // ■■■■■■■ 1: ■■■■■■ ■■■■■■■■ ■■ 10 ■■■■■■ ■■■■■■
        int[] numbers = new int[10];

        // Step 2: Get input from user
        // ■■■■■■■ 2: ■■■■■■ ■■■■■■■■ ■■ ■■■■■■■■■
        Console.WriteLine("Please enter 10 integer numbers:");
        Console.WriteLine("■■ ■■■■ ■■■■ 10 ■■■■■■ ■■■■■■:");

        for (int i = 0; i < 10; i++)
        {
            Console.Write($"Enter number {i + 1}: ");
            numbers[i] = int.Parse(Console.ReadLine());
        }

        // Step 3: Initialize counters for odd and even numbers
        // ■■■■■■■ 3: ■■■■■ ■■■■■■■ ■■■■■■■ ■■■■■■■ ■■■■■■■■
        int evenCount = 0;  // Counter for even numbers - ■■■■ ■■■■■■■■ ■■■■■■■
        int oddCount = 0;   // Counter for odd numbers - ■■■■ ■■■■■■■■ ■■■■■■■

        // Step 4: Use foreach loop to count odd and even numbers
        // ■■■■■■■ 4: ■■■■■■■■ ■■■■ foreach ■■■ ■■■■■■■■
        foreach (int number in numbers)
        {
            // Step 5: Check if number is even or odd
            // ■■■■■■■ 5: ■■■ ■■■ ■■■ ■■■■■■ ■■■■ ■■ ■■■■
            if (number % 2 == 0)
            {
```

```csharp
                    evenCount++;  // Increment even counter - ■■■■■ ■■■■ ■■■■■■■
                }
                else
                {
                    oddCount++;   // Increment odd counter - ■■■■■ ■■■■ ■■■■■■■
                }
            }

            // Step 6: Display the results
            // ■■■■■■■ 6: ■■■ ■■■■■■■■
            Console.WriteLine("\n===== Results / ■■■■■■■■ =====");
            Console.WriteLine($"Number of Even numbers: {evenCount}");
            Console.WriteLine($"■■■ ■■■■■■■■ ■■■■■■■■: {evenCount}");
            Console.WriteLine($"Number of Odd numbers: {oddCount}");
            Console.WriteLine($"■■■ ■■■■■■■■ ■■■■■■■■: {oddCount}");

            Console.ReadKey();
        }
    }
```

Example Output / ■■■■ ■■■ ■■■■■■■■:

```
Please enter 10 integer numbers:
■■ ■■■■ ■■■■ 10 ■■■■■■ ■■■■■:
Enter number 1: 5
Enter number 2: 8
Enter number 3: 12
Enter number 4: 7
Enter number 5: 20
Enter number 6: 15
Enter number 7: 3
Enter number 8: 10
Enter number 9: 9
Enter number 10: 14

===== Results / ■■■■■■■■ =====
Number of Even numbers: 5
```

■■■ ■■■■■■■■ ■■■■■■■■: 5

Number of Odd numbers: 5

■■■ ■■■■■■■■ ■■■■■■■■: 5

Question 3 (Part 3) (20 Marks)

██████ ██████ (█████ ██████) (20 █████)

Question: What is the output on screen of the following code?

██████: ██ ██ ███████ ███ ██████ █████ ███████

```
for (int i = 0; i < 10; i++)
{
    if (i == 2 || i == 6)  { continue; }
    if (i == 4 && i == 8)  { break;    }
    Console.Write(i);
}
```

Solution Explanation / ███ █████:

Let's trace through the code step by step:
████ █████ █████ ████ █████:

Understanding the keywords / ███ ███████ ██████████:
• **continue**: Skip the rest of the current iteration and go to next iteration
████ ████ ███████ ██████ █████████ ███████ ██████

• **break**: Exit the loop completely
██████ ██ ██████ ████ ████

Step-by-step execution / ███████ ████ █████:
• i = 0: No conditions met, print 0 → Output: **0**
• i = 1: No conditions met, print 1 → Output: **01**
• i = 2: First condition (i == 2) is TRUE, **continue** → Skip printing, Output: **01**
• i = 3: No conditions met, print 3 → Output: **013**
• i = 4: First condition FALSE (4 ≠ 2 and 4 ≠ 6), Second condition FALSE (4 == 4 but 4 ≠ 8, so 4 == 4 && 4 == 8 is FALSE), print 4 → Output: **0134**
• i = 5: No conditions met, print 5 → Output: **01345**
• i = 6: First condition (i == 6) is TRUE, **continue** → Skip printing, Output: **01345**
• i = 7: No conditions met, print 7 → Output: **013457**
• i = 8: First condition FALSE (8 ≠ 2 and 8 ≠ 6), Second condition FALSE (8 ≠ 4, so 8 == 4 && 8 == 8 is FALSE), print 8 → Output: **0134578**
• i = 9: No conditions met, print 9 → Output: **01345789**

Important Note / ██████ ████:
The second condition (i == 4 && i == 8) will NEVER be true because a variable cannot equal 4 AND 8 at the same time!

■■■■■ ■■■■■■■ (i == 4 && i == 8) ■■ ■■■■ ■■■■■■■ ■■■■■ ■■■ ■■■■■■■ ■■ ■■■■ ■■ ■■■■■■ 4 ■ 8 ■■ ■■■ ■■■■■!
So the **break** statement will never execute.
■■■■ ■■■■ **break** ■■ ■■■■ ■■■■■.

Final Answer / ■■■■■■■ ■■■■■■■■■:

The correct answer is: **c. 01345789**

■■■■■■■ ■■■■■■■ ■■: **c. 01345789**

The program prints all numbers from 0 to 9, except 2 and 6 (which are skipped by continue).
■■■■■■■■■ ■■■■ ■■■■ ■■■■■■■ ■■ 0 ■■■ 9■ ■■ ■■■ 2 ■ 6 (■■■■ ■■■ ■■■■■■ ■■■■■■ continue).

Note: This solution is designed to be beginner-friendly with detailed explanations in both English and Arabic.
■■■■■■■: ■■■ ■■■■ ■■■■ ■■■■■ ■■■■■■■ ■■■■■■■■ ■■ ■■■■■■ ■■■■■■ ■■■■■■■■■■ ■■■■■■■.