



Mansoura University
Faculty of Computers and Information

Structured Programming

(CS1001)

Lab Manual

Student's Edition

Dr. Mohamed Handosa

Fall 2024

Contents

1	Hello World	1
1.1	Objective	1
1.2	Problem	1
2	Type Conversion	3
2.1	Objective	3
2.2	Problem	3
3	Arithmetic Operations	5
3.1	Objective	5
3.2	Problem	5
4	Conditional Statements	7
4.1	Objective	7
4.2	Problem	7

5	Nested Conditional Statements	9
5.1	Objective	9
5.2	Problem	9
6	Looping Statements	11
6.1	Objective	11
6.2	Problem	11
7	Nested Looping Statements	13
7.1	Objective	13
7.2	Problem	13
8	One-Dimensional Arrays	15
8.1	Objective	15
8.2	Problem	15
9	Two-Dimensional Arrays	17
9.1	Objective	17
9.2	Problem	17
10	Method Definition and Call	19
10.1	Objective	19
10.2	Problem	19

11 Passing by Reference	21
11.1 Objective	21
11.2 Problem	21
 12 Arrays, Loops, and Methods	 23
12.1 Objective	23
12.2 Problem	23

Lab 1

Hello World

1.1 Objective

This lab aims to introduce students to the integrated development environment named Microsoft Visual Studio.

1.2 Problem

Write a program that defines two variables. The first variable is of type `string` and it should store your name. The second variable is of type `int` and it should store your age. Afterward, print the values of the two variables to the screen.

Lab 2

Type Conversion

2.1 Objective

This lab aims to train students on using type casting and type conversion.

2.2 Problem

Write a program that takes a floating-point number as input and prints its rounded value to the screen.

Lab 3

Arithmetic Operations

3.1 Objective

This lab aims to train students to analyze a problem and develop a program to solve it. This should take place by determining the inputs and the outputs from the problem description. Afterward, determine the processing steps required to generate the outputs from the inputs.

3.2 Problem

Write a program that allows for performing the four arithmetic operations (i.e. addition, subtraction, multiplication, and division) on any two real numbers and prints the results on the screen.

Lab 4

Conditional Statements

4.1 Objective

This lab aims to train students on analyzing a problem, understand it, and determine the steps required to solve it pragmatically. Moreover, introduce the student to conditional statements, their role in a program, and when and how to use them.

4.2 Problem

Write a program for a retail company that allows a user to enter a unit price and the number of units. Afterward, the program should calculate the total price. If the total price is less than 1000 £, the program should print a bill with an amount due that is equal to the calculated total price. Otherwise, the program should print a bill with an amount due that is equal to the calculated total price after applying a discount of 10%.

Lab 5

Nested Conditional Statements

5.1 Objective

This lab aims to introduce students to nested conditional statements, help students understand when to use them, and train students on using them.

5.2 Problem

The following program takes a number as input and outputs whether it is even or odd as well as whether it is negative or positive.

```
1  using System;
2  namespace Lab05
3  {
4      class Program
5      {
6          static void Main()
7          {
8              Console.Write("Enter a number: ");
9              int n = int.Parse(Console.ReadLine());
```

```
10
11     if(n % 2 == 0 && n < 0)
12     {
13         Console.WriteLine("Even and negative");
14     }
15
16     if (n % 2 == 0 && n >= 0)
17     {
18         Console.WriteLine("Even and positive");
19     }
20
21     if (n % 2 != 0 && n < 0)
22     {
23         Console.WriteLine("Odd and negative");
24     }
25
26     if (n % 2 != 0 && n >= 0)
27     {
28         Console.WriteLine("Odd and positive");
29     }
30 }
31 }
32 }
```

Every time the previous program is run, it requires the processor to perform 8 different comparisons. Rewrite that program using nested `if` statements to reduce the number of required comparisons to 2 comparisons only, whenever it is run.

Lab 6

Looping Statements

6.1 Objective

This lab aims to train students on using loops to print simple shapes.

6.2 Problem

Using loops, write a program to print the following shape on screen:

```
1  *****
2  *
3  *
4  *
5  *
6  *
7  *
8  *
```

In order to print output to the screen, you can only use:

- `System.Console.Write('*')`
- and `System.Console.WriteLine()`

Lab 7

Nested Looping Statements

7.1 Objective

This lab aims to train students on using nested loops to print complex shapes.

7.2 Problem

Using nested loops, write a program to print the following shape on screen:

```
1  *****
2  ****  ****
3  ***   ***
4  **    **
5  *      *
6  *      *
7  **     **
8  ***    ***
9  ****   ****
10 *****
```

In order to print output to the screen, you can only use:

- `System.Console.Write('*')`
- `System.Console.Write(' ')`
- `System.Console.WriteLine()`

Lab 8

One-Dimensional Arrays

8.1 Objective

This lab aims to train students to manipulate the elements of a one-dimensional array.

8.2 Problem

Write a program that takes a set of integer values as input and prints them to the screen in reverse order.

Lab 9

Two-Dimensional Arrays

9.1 Objective

This lab aims to train students to manipulate the elements of a two-dimensional array using nested loops.

9.2 Problem

Complete the following program by providing the missing code in the positions designated by the comments.

```
1 using System;
2 namespace Lab11
3 {
4     class Program
5     {
6         static void Main()
7         {
8             int[,] a = {
9                 { 1, 2, 3, 4 },
10                { 5, 6, 7, 8 },
11                { 9, 10, 11, 12 } };
12
13            Console.WriteLine("Matrix: ");
14
15            // MISSING CODE (print array a)
16
17            int[,] b = new int[4,3];
18
19            // MISSING CODE (transpose a and save it to b)
20
21            Console.WriteLine("Transposed matrix: ");
22
23            // MISSING CODE (print array b)
24        }
25    }
26 }
```

The output of the program should be as follows:

```
1 Matrix:
2 1      2      3      4
3 5      6      7      8
4 9      10     11     12
5 Transposed matrix:
6 1      5      9
7 2      6      10
8 3      7      11
9 4      8      12
```


Lab 10

Method Definition and Call

10.1 Objective

This lab aims to train students to read a method definition and understand it. It also checks the student's ability to understand the execution order of called methods. Students should be able to use methods in such a way that fulfills program specifications.

10.2 Problem

Write a program that prints the following on the screen:

```
1 Hello
2 Hello
3 Hello
4 Hi
5 Hello
6 Hello
7 Hi
```

This should be done by calling only the `Method1` and `Method2` with the correct number of times and order from within the `Main` method without adding any other code.

```
1 using System;
2
3 namespace Lab08
4 {
5     class Program
6     {
7         static void Main(string[] args)
8         {
9             // MISSING CODE
10        }
11
12        static void Method1()
13        {
14            Method2();
15            Console.WriteLine("Hi");
16        }
17        static void Method2()
18        {
19            Console.WriteLine("Hello");
20        }
21    }
22 }
```

Lab 11

Passing by Reference

11.1 Objective

This lab aims to train students to use pass by reference. The student should understand the difference between passing by value and passing by reference. Moreover, the student should recognize the benefits of passing by reference (e.g. returning more than one value from a function).

11.2 Problem

Write a program to perform the four basic arithmetic operations on two numeric values. The `Main` method should input the two values before calling another method named `Calc` and pass the two values to it as arguments. Afterward, the `Calc` method should calculate the summation, difference, multiplication, and division of the passed values and return the results to the `Main` method that should print the four results on the screen.

Lab 12

Arrays, Loops, and Methods

12.1 Objective

This lab aims to train students on passing an array as a parameter to a method. It also aims to train students to manipulate array elements using loops.

12.2 Problem

Complete the following program by providing a definition of the `IndexOf` method that takes an array and a value as parameters. The method should search for the value in the array. It should return the index of the value if found in the array. Otherwise, it should return `-1`.

```
1  using System;
2
3  namespace Lab12
4  {
5      class Program
6      {
7          static void Main()
8          {
9              int[] x = { 12, 7, 1, 4, 18, 26, 30 };
10
11              Console.WriteLine("Index of 26 = {0}",
12                               IndexOf(x, 26));
13
14              Console.WriteLine("Index of 81 = {0}",
15                               IndexOf(x, 81));
16          }
17      }
18  }
```

The program output should be as follows:

```
1  Index of 26 = 5
2  Index of 81 = -1
```