

# OOP LAB TASK # 11

Name: Shahmeer khan.

ClassID: 106278.

Student-ID:12113.

## Task:

### Question no. 1:

### Inputted Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab_Task
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("OOP LAB TASK 11:");
            Console.WriteLine("Question no. 1:");
            Human function = new Human();
            function.Total_Distance(10, 12);
            Console.ReadKey();
        }
    }
    class Human
    {
        public void Total_Distance(int Walk, int Run)
        {
            for (int i = 0; i <= Walk; i++)
            {
                Console.WriteLine("\nRunning at a speed of " + Run + " km/h.\n Walking..
Steps covered: " + i +
                "\n Distance covered: " + Walk + "\n");
            }
        }
    }
}
```

```

0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        Console.WriteLine("OOP LAB TASK 11:");
        Console.WriteLine("Question no. 1:");
        Human function = new Human();
        function.Total_Distance(10, 12);
        Console.ReadKey();
    }
}

2 references
class Human
{
    1 reference
    public void Total_Distance(int Walk, int Run)
    {
        for (int i = 0; i <= Walk; i++)
        {
            Console.WriteLine("\nRunning at a speed of " + Run + " km/h.\n Walking.. Steps covered: " + i +
                "\n Distance covered: " + Walk + "\n");
        }
    }
}

```

## Output:

C:\Users\hp\source\repos\Lab Task\Lab Task\bin\Debug\Lab Task.exe

```

OOP LAB TASK 11:
Question no. 1:

Running at a speed of 12 km/h.
Walking.. Steps covered: 0
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 1
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 2
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 3
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 4
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 5
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 6
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 7
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 8
Distance covered: 10

Running at a speed of 12 km/h.
Walking.. Steps covered: 9

```

## Question no. 2:

### Inputted Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab_Task
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("OOP LAB TASK 11:");
            Console.WriteLine("Question no. 2:");
            Console.Write("Enter Starting Point: ");
            string Start = Console.ReadLine();
            Console.Write("\nEnter Stopping Point: ");
            string Stop = Console.ReadLine();
            Train obj = new Train();
            Console.WriteLine("\nObject 1 used:");
            obj.EngineWork(Start, Stop);
            string Work;
            Work = Start;
            Start = Stop;
            Stop = Work;
            Console.WriteLine("\n-----");
            ReverseTrain obj2 = new ReverseTrain();
            Console.WriteLine("\nObject 2 used:");
            obj2.EngineWork(Start, Stop); //EngineWork method easilly called because of
inheritance
            Console.ReadKey();
        }
    }
    class Train
    {
        public void EngineWork(string Source, string Destination)
        {
            Console.WriteLine("Starting-Point: " + Source);
            Console.WriteLine("Ending-Point: " + Destination);
        }
    }
    class ReverseTrain : Train
    {
    }
}
```

Program.cs

Lab Task

Lab\_Task.Program

Main(string[] args)

```
11 0 references
12 static void Main(string[] args)
13 {
14     Console.WriteLine("OOP LAB TASK 11:");
15     Console.WriteLine("Question no. 2:");
16     Console.WriteLine("Enter Starting Point: ");
17     string Start = Console.ReadLine();
18     Console.WriteLine("\nEnter Stopping Point: ");
19     string Stop = Console.ReadLine();
20     Train obj = new Train();
21     Console.WriteLine("\nObject 1 used:");
22     obj.EngineWork(Start, Stop);
23     string Work;
24     Work = Start;
25     Start = Stop;
26     Stop = Work;
27     Console.WriteLine("\n-----");
28     ReverseTrain obj2 = new ReverseTrain();
29     Console.WriteLine("\nObject 2 used:");
30     obj2.EngineWork(Start, Stop); //EngineWork method easilly called because of inheritance
31     Console.ReadKey();
32 }
33 3 references
34 class Train
35 {
36     2 references
37     public void EngineWork(string Source, string Destination)
38     {
39         Console.WriteLine("Starting-Point: " + Source);
40         Console.WriteLine("Ending-Point: " + Destination);
41     }
42 }
```

90 % No issues found Ln: 25 Ch: 26 SPC CRLF

Output

Show output from: Debug

'Lab Task.exe' (CLR v4.0.30319: DefaultDomain): Loaded 'C:\Windows\Microsoft.Net\assembly\GAC\_32\mscorlib\v4.0.4.0.0\_\_b77a5c561934e089\mscorlib.dll'. Skipped loading 'Lab Task.exe' (CLR v4.0.30319: DefaultDomain): Loaded 'C:\Users\hp\source\repos\Lab Task\Lab Task\bin\Debug\Lab Task.exe'. Symbols loaded. The program '[596] Lab Task.exe' has exited with code -1073741510 (0xc00013a).

Program.cs

Lab Task

Lab\_Task.Program

Main(string[] args)

```
17 Console.WriteLine("\nEnter Stopping Point: ");
18 string Stop = Console.ReadLine();
19 Train obj = new Train();
20 Console.WriteLine("\nObject 1 used:");
21 obj.EngineWork(Start, Stop);
22 string Work;
23 Work = Start;
24 Start = Stop;
25 Stop = Work;
26 Console.WriteLine("\n-----");
27 ReverseTrain obj2 = new ReverseTrain();
28 Console.WriteLine("\nObject 2 used:");
29 obj2.EngineWork(Start, Stop); //EngineWork method easilly called because of inheritance
30 Console.ReadKey();
31 }
32 }
33 3 references
34 class Train
35 {
36     2 references
37     public void EngineWork(string Source, string Destination)
38     {
39         Console.WriteLine("Starting-Point: " + Source);
40         Console.WriteLine("Ending-Point: " + Destination);
41     }
42 }
43 2 references
44 class ReverseTrain : Train
45 {
46 }
```

90 % No issues found Ln: 25 Ch: 26 SPC CRLF

Output

Show output from: Debug

'Lab Task.exe' (CLR v4.0.30319: DefaultDomain): Loaded 'C:\Windows\Microsoft.Net\assembly\GAC\_32\mscorlib\v4.0.4.0.0\_\_b77a5c561934e089\mscorlib.dll'. Skipped loading 'Lab Task.exe' (CLR v4.0.30319: DefaultDomain): Loaded 'C:\Users\hp\source\repos\Lab Task\Lab Task\bin\Debug\Lab Task.exe'. Symbols loaded. The program '[596] Lab Task.exe' has exited with code -1073741510 (0xc00013a).

## Output:

C:\Users\hp\source\repos\Lab Task\Lab Task\bin\Debug\Lab Task.exe

```
OOP LAB TASK 11:
Question no. 2:
Enter Starting Point: This is Starting Point

Enter Stopping Point: This is Ending Point

Object 1 used:
Starting-Point: This is Starting Point
Ending-Point: This is Ending Point
```

```
-----
Object 2 used:
Starting-Point: This is Ending Point
Ending-Point: This is Starting Point
```

## Question no. 3:

### Inputted Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab_Task
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("OOP LAB TASK 11:");
            Console.WriteLine("Question no. 3:");
            Console.Write("Enter Starting Point: ");
            string Start = Console.ReadLine();
            Console.Write("\nEnter Stopping Point: ");
            string Stop = Console.ReadLine();
            Train obj = new Train();
            Console.WriteLine("\nObject 1 used:");
            obj.EngineWork(Start, Stop);
            Console.WriteLine("\n-----");
            ReverseTrain obj2 = new ReverseTrain();
```

```

        Console.WriteLine("\nEnter journey Time: ");
        int JT = int.Parse(Console.ReadLine());
        Console.WriteLine("\nObject 2 used:");
        obj2.EngineWork(Start, Stop, JT); //EngineWork method easily called because
of inheritance
        Console.ReadKey();
    }
}
class Train
{
    public string Start;
    public string End;
    public void EngineWork(string Source, string Destination)
    {
        this.Start = Source;
        this.End = Destination;
        Console.WriteLine("Starting Point: " + Start);
        Console.WriteLine("Ending Point: " + End);
    }
}
class ReverseTrain : Train
{
    public void EngineWork(string Source, string Destination, int JourneyTime)
    {
        // base.EngineWork(Source, Destination);
        this.Start = Source;
        this.End = Destination;
        Console.WriteLine("Starting Point: " + Start);
        Console.WriteLine("Ending Point: " + End);
        Console.WriteLine("Total Journey Time: " + JourneyTime + " hrs");
    }
}
}

```

The screenshot shows the Visual Studio IDE with the 'Program.cs' file open. The code is displayed in a light blue theme. The bottom status bar indicates '90 %' and 'No issues found'. The bottom toolbar includes 'Locals', 'Search (Ctrl+E)', 'Search Depth', and 'Breakpoints'.

```

12      {
13          Console.WriteLine("OOP LAB TASK 11:");
14          Console.WriteLine("Question no. 3:");
15          Console.WriteLine("Enter Starting Point: ");
16          string Start = Console.ReadLine();
17          Console.WriteLine("Enter Stopping Point: ");
18          string Stop = Console.ReadLine();
19          Train obj = new Train();
20          Console.WriteLine("\nObject 1 used:");
21          obj.EngineWork(Start, Stop);
22          Console.WriteLine("\n-----");
23          ReverseTrain obj2 = new ReverseTrain();
24          Console.WriteLine("\nEnter journey Time: ");
25          int JT = int.Parse(Console.ReadLine());
26          Console.WriteLine("\nObject 2 used:");
27          obj2.EngineWork(Start, Stop, JT); //EngineWork method easily called because of inheritance
28          Console.ReadKey();
29      }
30  }
31  class Train
32  {
33      public string Start;
34      public string End;

```

Program.cs

Lab Task

Lab\_Task.Program

```
30 }
31 3 references
32 class Train
33 {
34     public string Start;
35     public string End;
36     1 reference
37     public void EngineWork(string Source, string Destination)
38     {
39         this.Start = Source;
40         this.End = Destination;
41         Console.WriteLine("Starting Point: " + Start);
42         Console.WriteLine("Ending Point: " + End);
43     }
44 }
45 2 references
46 class ReverseTrain : Train
47 {
48     1 reference
49     public void EngineWork(string Source, string Destination, int JourneyTime)
50     {
51         // base.EngineWork(Source, Destination);
52         this.Start = Source;
53         this.End = Destination;
54         Console.WriteLine("Starting Point: " + Start);
55     }
56 }
```

90 % No issues found

Locals

Search (Ctrl+E) Search Depth:

Program.cs

Lab Task

Lab\_Task.Program

Main(string[] args)

```
35 1 reference
36 public void EngineWork(string Source, string Destination)
37 {
38     this.Start = Source;
39     this.End = Destination;
40     Console.WriteLine("Starting Point: " + Start);
41     Console.WriteLine("Ending Point: " + End);
42 }
43 2 references
44 class ReverseTrain : Train
45 {
46     1 reference
47     public void EngineWork(string Source, string Destination, int JourneyTime)
48     {
49         // base.EngineWork(Source, Destination);
50         this.Start = Source;
51         this.End = Destination;
52         Console.WriteLine("Starting Point: " + Start);
53         Console.WriteLine("Ending Point: " + End);
54         Console.WriteLine("Total Journey Time: " + JourneyTime + " hrs");
55     }
56 }
```

90 % No issues found

Locals

Search (Ctrl+E) Search Depth:

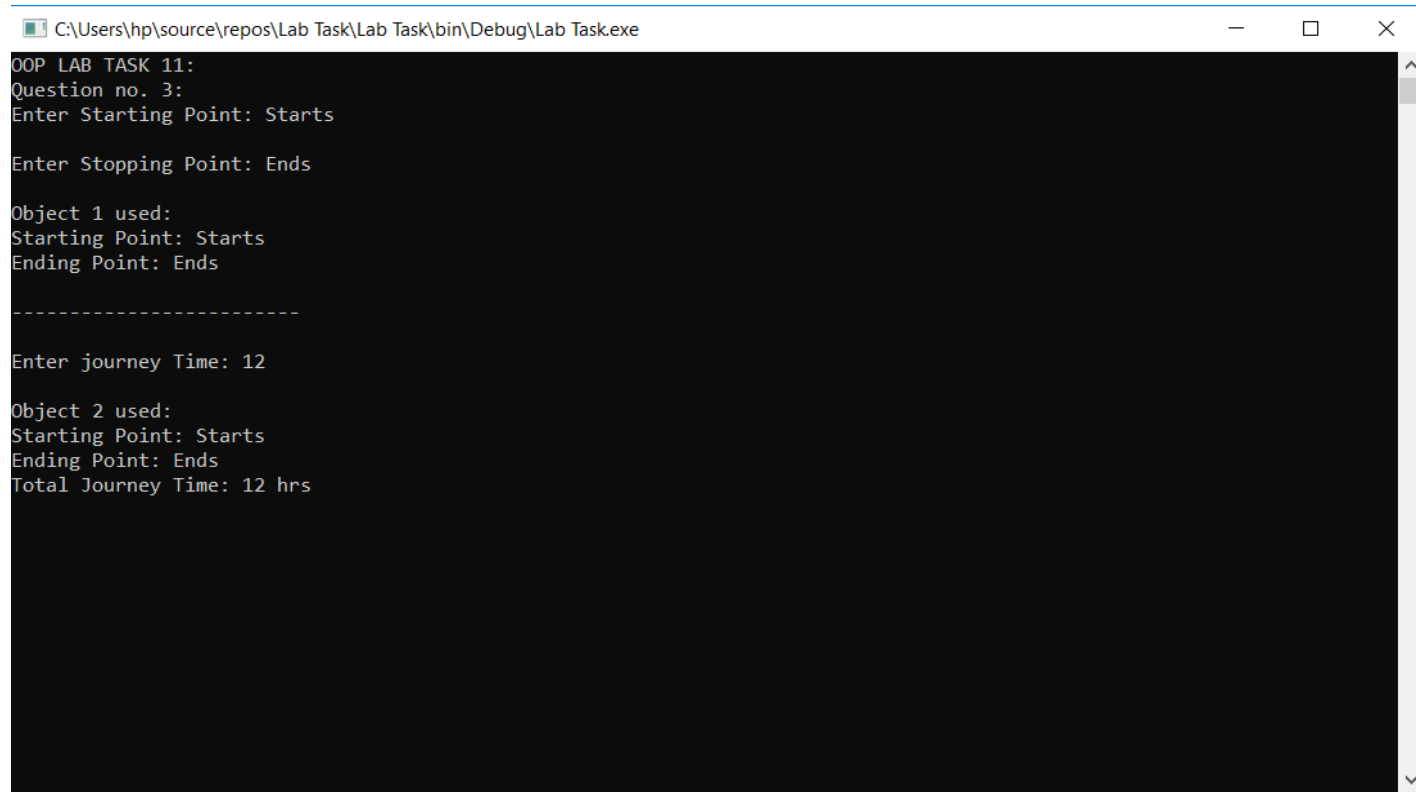
Name	Value	Type
------	-------	------

Breakpoint

New

Name	Lat
------	-----

## Output:



```
C:\Users\hp\source\repos\Lab Task\Lab Task\bin\Debug\Lab Task.exe
OOP LAB TASK 11:
Question no. 3:
Enter Starting Point: Starts
Enter Stopping Point: Ends
Object 1 used:
Starting Point: Starts
Ending Point: Ends
-----
Enter journey Time: 12
Object 2 used:
Starting Point: Starts
Ending Point: Ends
Total Journey Time: 12 hrs
```

## Question no. 4:

### Inputted Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab_Task
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("OOP LAB TASK 11:");
            Console.WriteLine("Question no. 4:");
            Console.WriteLine("\nEnter the distance, in Kilometers, covered in Journey A:");
            int a = int.Parse(Console.ReadLine());
            Console.WriteLine("\nEnter the distance, in Kilometers, covered in Journey B:");
            int b = int.Parse(Console.ReadLine());
            ReverseTrain value1 = new ReverseTrain(a);
            ReverseTrain value2 = new ReverseTrain(b);
            ReverseTrain value3 = new ReverseTrain();
            value3 = value1 + value2;
            value3.TotalDistance();
        }
    }
}
```

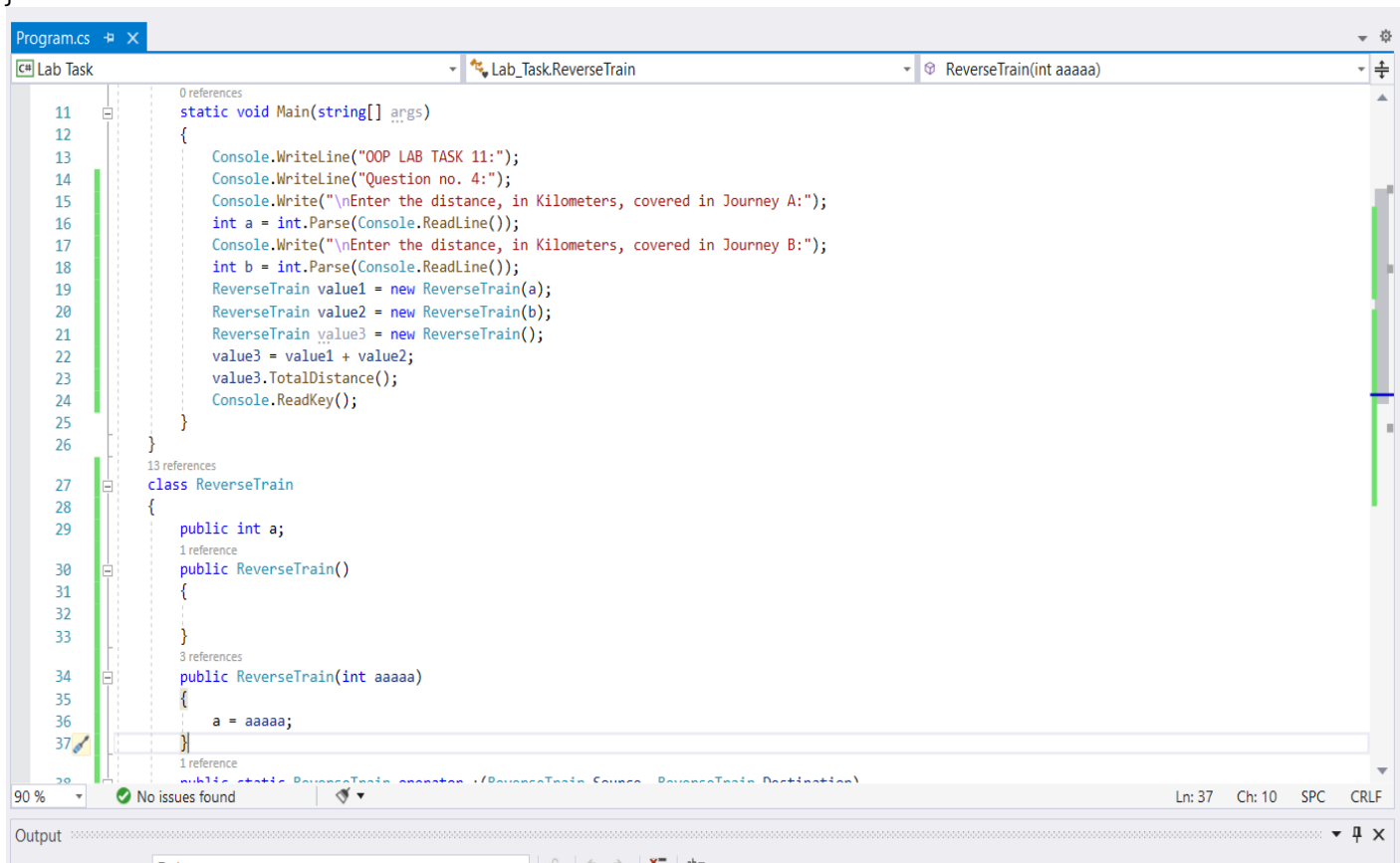


```

        Console.ReadKey();
    }
}
class ReverseTrain
{
    public int a;
    public ReverseTrain()
    {

    }
    public ReverseTrain(int aaaaa)
    {
        a = aaaaa;
    }
    public static ReverseTrain operator +(ReverseTrain Source, ReverseTrain
Destination)
    {
        // base.EngineWork(Source, Destination);
        ReverseTrain rev3 = new ReverseTrain(0);
        rev3.a = Source.a + Destination.a;
        return rev3;
    }
    public void TotalDistance()
    {
        Console.WriteLine("Total Distance: {0} Kilometers", a);
    }
}
}

```



```
Program.cs  x
Lab Task  Lab_Task.ReverseTrain  ReverseTrain(int aaaaa)
26  }
27  13 references
28  class ReverseTrain
29  {
30      public int a;
31      1 reference
32      public ReverseTrain()
33      {
34      }
35      3 references
36      public ReverseTrain(int aaaaa)
37      {
38          a = aaaaa;
39      }
40      1 reference
41      public static ReverseTrain operator +(ReverseTrain Source, ReverseTrain Destination)
42      {
43          // base.EngineWork(Source, Destination);
44          ReverseTrain rev3 = new ReverseTrain(0);
45          rev3.a = Source.a + Destination.a;
46          return rev3;
47      }
48      1 reference
49      public void TotalDistance()
50      {
51          Console.WriteLine("Total Distance: {0} Kilometers", a);
52      }
53  }
```

90 % No issues found Ln: 37 Ch: 10 SPC CRLF

## Output:

```
C:\Users\hp\source\repos\Lab Task\Lab Task\bin\Debug\Lab Task.exe
OOP LAB TASK 11:
Question no. 4:

Enter the distance, in Kilometers, covered in Journey A:12
Enter the distance, in Kilometers, covered in Journey B:54
Total Distance: 66 Kilometers
```