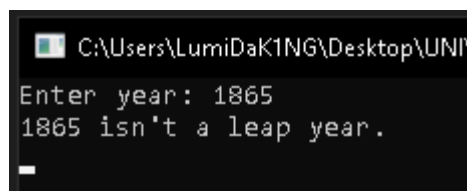


Chapter 11. Creating and Using Objects

1. Write a program, which reads from the console a year and checks if it is a leap year.

```
using System;

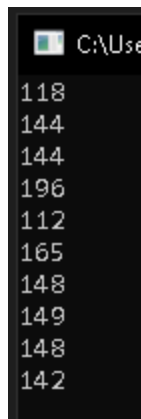
namespace ex1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter year: ");
            int year = Int32.Parse(Console.ReadLine());
            if (((year % 4) == 0) && ((year % 100) != 0) || ((year % 400) == 0))
                Console.WriteLine("{0} is leap year.", year);
            else
                Console.WriteLine("{0} isn't a leap year.", year);
            Console.ReadKey();
        }
    }
}
```



2. Write a program, which generates and prints on the console 10 random numbers in the range [100, 200].

```
using System;

namespace ex2
{
    class Program
    {
        static void Main(string[] args)
        {
            Random r = new Random();
            for (int i = 0; i < 10; i++)
                Console.WriteLine(r.Next(100, 201));
            Console.ReadKey();
        }
    }
}
```



A screenshot of a console window with a black background and white text. The window title bar shows a small icon and the text "C:\Use". The console displays 10 lines of random numbers generated by the program: 118, 144, 144, 196, 112, 165, 148, 149, 148, and 142.

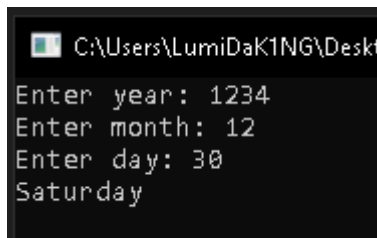
Line	Random Number
1	118
2	144
3	144
4	196
5	112
6	165
7	148
8	149
9	148
10	142

3. Write a program, which prints, on the console which day of the week is today.

```
using System;

namespace ex3
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter year: ");
            int year = Int32.Parse(Console.ReadLine());
            Console.Write("Enter month: ");
            int month = Int32.Parse(Console.ReadLine());
            Console.Write("Enter day: ");
            int day = Int32.Parse(Console.ReadLine());

            DateTime dateValue = new DateTime(year, month, day);
            Console.WriteLine(dateValue.ToString("dddd"));
            Console.ReadKey();
        }
    }
}
```



A screenshot of a Windows command prompt window. The title bar shows the file path "C:\Users\LumiDaK1NG\Desktop". The console text shows the program's execution: "Enter year: 1234", "Enter month: 12", "Enter day: 30", and finally "Saturday".

4. Write a program, which prints on the standard output the count of days, hours, and minutes, which have passes since the computer is started until the moment of the program execution. For the implementation use the class Environment.

5. Write a program which by given two sides finds the hypotenuse of a right triangle. Implement entering of the lengths of the sides from the standard input, and for the calculation of the hypotenuse use methods of the class Math.

```
using System;
```

```
namespace ex5
```

```
{
```

```
    class Program
```

```
    {
```

```
        static void Main(string[] args)
```

```
        {
```

```
            Console.Write("First side: ");
```

```
            int a = Int32.Parse(Console.ReadLine());
```

```
            Console.Write("Second side: ");
```

```
            int b = Int32.Parse(Console.ReadLine());
```

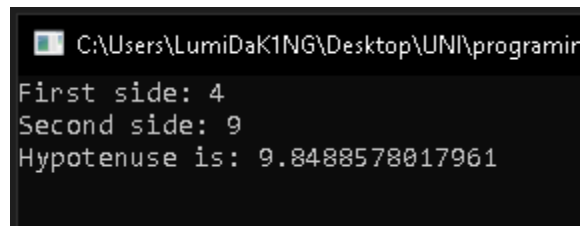
```
            Console.WriteLine("Hypotenuse is: " + Math.Sqrt(Math.Pow(a, 2) + Math.Pow(b, 2)));
```

```
            Console.ReadKey();
```

```
        }
```

```
    }
```

```
}
```



```
C:\Users\LumiDaK1NG\Desktop\UNI\programir
First side: 4
Second side: 9
Hypotenuse is: 9.8488578017961
```

6. Write a program which calculates the area of a triangle with the following given:

- three sides;
- side and the altitude to it;
- two sides and the angle between them in degrees.

```
using System;

namespace ex6
{
    class Program
    {
        public static void byThreeSides()
        {
            Console.WriteLine("\nEnter a: ");
            float a = float.Parse(Console.ReadLine());
            Console.WriteLine("Enter b: ");
            float b = float.Parse(Console.ReadLine());
            Console.WriteLine("Enter c: ");
            float c = float.Parse(Console.ReadLine());
            float p = (a + b + c) / 2;
            Console.WriteLine("S is: {0}", (float)(Math.Sqrt(p * (p - a) * (p - b) * (p - c))));
            Console.ReadLine();
        }

        public static void bySideAltitude()
        {
            Console.WriteLine("\nEnter a: ");
            float a = float.Parse(Console.ReadLine());
            Console.WriteLine("Enter h(a): ");
            float b = float.Parse(Console.ReadLine());
            Console.WriteLine("S is: {0}", (a * b) / 2);
            Console.ReadLine();
        }

        public static void byTwoSidesAngle()
        {
            Console.WriteLine("\nEnter a: ");
            float a = float.Parse(Console.ReadLine());
            Console.WriteLine("Enter b: ");
            float b = float.Parse(Console.ReadLine());
            Console.WriteLine("Enter sine: ");
            float c = float.Parse(Console.ReadLine());
            Console.WriteLine("S is: {0}", (a * b * Math.Sin(c)) / 2);
            Console.ReadLine();
        }

        static void Main(string[] args)
```

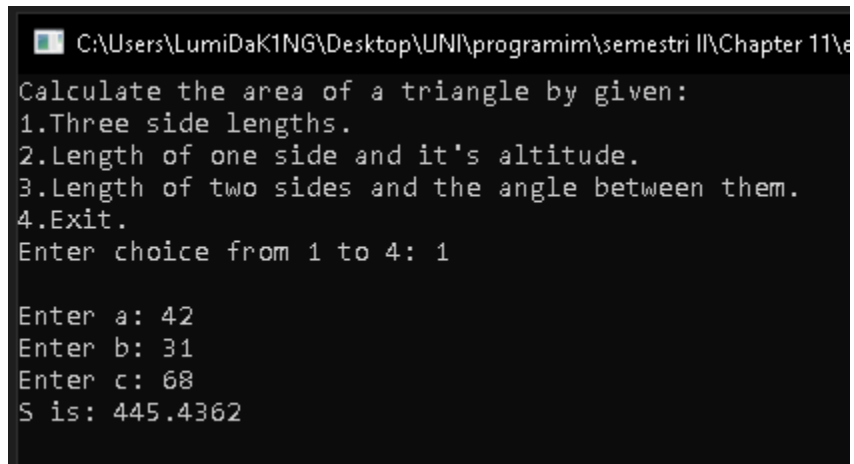
```

{
    byte choice;
    do
    {
        Console.Clear();
        Console.WriteLine("Calculate the area of a triangle by given:");
        Console.WriteLine("1.Three side lengths.");
        Console.WriteLine("2.Length of one side and it's altitude.");
        Console.WriteLine("3.Length of two sides and the angle between them.");
        Console.WriteLine("4.Exit.");
        Console.Write("Enter choice from 1 to 4: ");
        choice = System.Convert.ToByte(Console.ReadLine());

        switch (choice)
        {
            case 1: byThreeSides(); break;
            case 2: bySideAltitude(); break;
            case 3: byTwoSidesAngle(); break;
        }

    } while (choice != 4);
    Console.ReadKey();
}
}
}

```



```

C:\Users\LumiDaK1NG\Desktop\UNI\programim\semestri II\Chapter 11\
Calculate the area of a triangle by given:
1.Three side lengths.
2.Length of one side and it's altitude.
3.Length of two sides and the angle between them.
4.Exit.
Enter choice from 1 to 4: 1

Enter a: 42
Enter b: 31
Enter c: 68
S is: 445.4362

```

7. Define your own namespace `CreatingAndUsingObjects` and place in it two classes `Cat` and `Sequence`, which we used in the examples of the current chapter. Define one more namespace and make a class, which calls the classes `Cat` and `Sequence`, in it.

8. Write a program which creates 10 objects of type `Cat`, gives them names `CatN`, where `N` is a unique serial number of the object, and in the end call the method `SayMiau()` for each of them. For the implementation use the namespace `CreatingAndUsingObjects`.

9. Write a program, which calculates the count of workdays between the current date and another given date after the current (inclusive). Consider that workdays are all days from Monday to Friday, which are not public holidays, except when Saturday is a working day. The program should keep a list of predefined public holidays, as well as a list of predefined working Saturdays.

```
using System;

namespace ex9
{
    class Program
    {
        static void Main(string[] args)
        {
            DateTime[] holidays = new DateTime[]
            {
                new DateTime(2014, 12, 24),
                new DateTime(2014, 12, 25),
                new DateTime(2015, 01, 01),
                new DateTime(2015, 01, 02),
                new DateTime(2015, 03, 02),
                new DateTime(2015, 03, 03),
                new DateTime(2015, 04, 10),
                new DateTime(2015, 04, 13),
                new DateTime(2015, 05, 01),
                new DateTime(2015, 05, 06),
                new DateTime(2015, 09, 21),
                new DateTime(2015, 09, 22),
                new DateTime(2015, 12, 24),
                new DateTime(2015, 12, 25),
            };

            DateTime[] workSaturdays = new DateTime[]
            {
                new DateTime(2015, 01, 24),
                new DateTime(2015, 03, 21),
                new DateTime(2015, 09, 12),
                new DateTime(2015, 12, 12),
            };

            int workingDays = 0;

            Console.WriteLine("Enter end date (YYYY/MM/DD): ");
            DateTime endDate = System.Convert.ToDateTime(Console.ReadLine());
            DateTime now = DateTime.Now;

            do
            {
                now = now.AddDays(1);

                if ((now.DayOfWeek >= DayOfWeek.Monday) && (now.DayOfWeek <=
DayOfWeek.Friday))
                    workingDays++;

                foreach (var i in holidays)
                    if (i.Date == now.Date)
                        workingDays--;
            }
        }
    }
}
```

```

        foreach (var i in workSaturdays)
            if (i.Date == now.Date)
                workingDays++;

    } while (now.Date != endDate.Date);

    Console.WriteLine("{0} working days.", workingDays);
    Console.ReadKey();
}
}
}

```



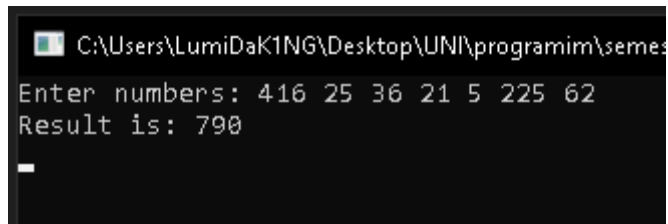
10. You are given a sequence of positive integer numbers given as string of numbers separated by a space. Write a program, which calculates their sum. Example: "43 68 9 23 318" à 461.

```
using System;

namespace ex10
{
    class Program
    {
        static void Main(string[] args)
        {
            int result = 0;
            Console.Write("Enter numbers: ");
            string inputNumbers = Console.ReadLine();
            string[] splitNumbers = inputNumbers.Split(' ');

            for (int i = 0; i < splitNumbers.Length; i++)
                result += Convert.ToInt32(splitNumbers[i]);

            Console.WriteLine("Result is: {0}", result);
            Console.ReadKey();
        }
    }
}
```



```
C:\Users\LumiDaK1NG\Desktop\UNI\programim\seme...
Enter numbers: 416 25 36 21 5 225 62
Result is: 790
_
```

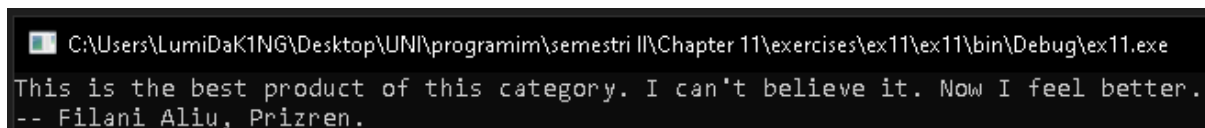
11. Write a program, which generates a random advertising message for some product. The message has to consist of laudatory phrase, followed by a laudatory story, followed by author (first and last name) and city, which are selected from predefined lists. For example, let's have the following lists:

using System;

namespace ex11

```
{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();
            string[] firstNames = new string[] { "Filani", "Hasani", "Remzia", "Fitija", "Fitimi" };
            string[] secondNames = new string[] { "Aliu", "Sabiti", "Fistku", "Bajraliu" };
            string[] cities = new string[] { "Ferizaj", "Prishte", "Prizren", "Peje", "Gjakove" };
            string[] reviews1 = new string[] { "This product is excelent.", "This is a great product.", "I use
this product all the time.", "This is the best product of this category." };
            string[] reviews2 = new string[] { "Now I feel better.", "I changed.", "It was a miracle.", "I can't
believe it. Now I feel better.", "Try it yourself. I'm very grateful." };

            Console.WriteLine("{0} {1}\n-- {2} {3}, {4}.", reviews1[rnd.Next(4)], reviews2[rnd.Next(5)],
firstNames[rnd.Next(1)], secondNames[rnd.Next(3)], cities[rnd.Next(4)]);
            Console.ReadKey();
        }
    }
}
```



```
C:\Users\LumiDaK1NG\Desktop\UNI\programim\semestri II\Chapter 11\exercises\ex11\ex11\bin\Debug\ex11.exe
This is the best product of this category. I can't believe it. Now I feel better.
-- Filani Aliu, Prizren.
```

12. * Write a program, which calculates the value of a given numeral expression given as a string.
The numeral expression consists of:

- real numbers, for example 5, 18.33, 3.14159, 12.6;
- arithmetic operations: +, -, *, / (with their standard priorities);
- mathematical functions: ln(x), sqrt(x), pow(x, y);
- brackets for changing the priorities of the operations: (and).

Note that the numeral expressions have priorities, for example the expression $-1 + 2 + 3 * 4 - 0.5$
 $= (-1) + 2 + (3 * 4) - 0.5 = 12.5$.

```
using System;
using System.Linq;

namespace ex12
{
    class Program
    {
        public static string equation;
        public static string[] equationArr = new string[50];
        public static string[] actions = new string[50];
        public static string[] numbers = new string[50];
        public static bool change = false;
        public static double result = 0;
        public static double temp = 0;
        public static byte depth = 0;
        public static byte brackets = 0;

        public static void removeNulls()
        {
            numbers = numbers.Where(x => !string.IsNullOrEmpty(x)).ToArray();
            actions = actions.Where(x => !string.IsNullOrEmpty(x)).ToArray();
        }

        static void Main(string[] args)
        {
            Console.WriteLine("Enter equation: ");
            equation = Console.ReadLine();
            equationArr = equation.Split(' ');

            for (int i = 0; i < equationArr.Length; i++)
            {
                if (change) actions[i] = equationArr[i];
                else numbers[i] = equationArr[i];
                change = change ? false : true;
            }
        }
    }
}
```

```

removeNulls();

do
{
    for (int i = 0; i < actions.Length; i++)
        if (i == 0)
            if (actions[i] == "*" || actions[i] == "/")
            {
                if (actions[i] == "*")
                    temp = Convert.ToDouble(numbers[0]) *
Convert.ToDouble(numbers[1]);
                else temp += Convert.ToDouble(numbers[0]) /
Convert.ToDouble(numbers[1]);

                numbers[0] = temp.ToString();
                actions[i] = numbers[1] = null;
                temp = 0;
                removeNulls();
            }
        else
            do
            {
                if (actions[i] == "*" || actions[i] == "/")
                {
                    if (actions[i] == "*") temp = Convert.ToDouble(numbers[i]) *
Convert.ToDouble(numbers[i + 1]);
                    else temp += Convert.ToDouble(numbers[i]) /
Convert.ToDouble(numbers[i + 1]);
                    numbers[i] = temp.ToString();
                    actions[i] = numbers[i + 1] = null;
                    temp = 0;
                    removeNulls();
                    if (depth > 0) depth--;
                }
                foreach (var s in actions)
                    if (s == "*" || s == "/") depth++;
            } while (depth != 0);

    for (int i = 0; i < actions.Length; i++)
        Console.WriteLine(actions[i]);

    for (int i = 0; i < numbers.Length; i++)
        Console.WriteLine(numbers[i]);

    Console.WriteLine("Continue {0} to +-", result);
    Console.ReadLine();

    for (int i = 0; i < actions.Length; i++)

```

```

        if (i == 0)
        {
            if (actions[i] == "+" || actions[i] == "-")
            {
                if (actions[i] == "+")
                {
                    numbers[0] = (Convert.ToDouble(numbers[0]) +
Convert.ToDouble(numbers[1])).ToString();
                }
                else numbers[0] = (Convert.ToDouble(numbers[0]) -
Convert.ToDouble(numbers[1])).ToString();

                actions[i] = numbers[1] = null;
                removeNulls();
                i--;
            }
        }
        else
        {
            do
            {
                if (actions[i] == "+" || actions[i] == "-")
                {
                    if (actions[i] == "+")
                    {
                        numbers[i] = (Convert.ToDouble(numbers[i]) +
Convert.ToDouble(numbers[i + 1])).ToString();
                    }
                    else numbers[i] = (Convert.ToDouble(numbers[i]) -
Convert.ToDouble(numbers[i + 1])).ToString();

                    actions[i] = numbers[i + 1] = null;
                    removeNulls();
                    if (depth > 0) depth--;
                }
            } while (depth != 0);
            foreach (var s in actions)
            {
                if (s == "+" || s == "-") depth++;
            }
        }
    } while (depth != 0);

    for (int i = 0; i < actions.Length; i++)
        Console.WriteLine("actions {0}", actions[i]);

    for (int i = 0; i < numbers.Length; i++)
        Console.WriteLine("numbers {0}", numbers[i]);
} while (numbers.Length > 1);

Console.WriteLine("Result is {0}", numbers);
Console.ReadKey();
}
}
}

```

C:\Users\LumiDaK1NG\Desktop\UNI\pro

Enter equation:

2+5=7

2+5=7

Continue 0 to +-
-

numbers 2+5=7

Result is 2+5=7