1. Write a program that reads a string, **reverse** it and prints it to the console. For example: "**introduction**"  "**noitcudortni**".

|  |
| --- |
| using System; |
|  | using System.Text; |
|  |  |
|  | namespace Chapter\_13\_Solution\_2 |
|  | { |
|  | class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | Console.Write("Enter text to reverse: "); |
|  | string input = Console.ReadLine(); |
|  |  |
|  | StringBuilder sb = new StringBuilder(); |
|  |  |
|  | for (int i = input.Length - 1; i >= 0; i--) |
|  | { |
|  | sb.Append(input[i]); |
|  | } |
|  |  |
|  | input = sb.ToString(); |
|  |  |
|  | Console.WriteLine("Reversed: \"{0}\"", input); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that **checks whether the parentheses are placed correctly** in an arithmetic expression. Example of expression with correctly placed brackets: **((a+b)/5-d)**. Example of an incorrect expression: **)(a+b))**.

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_3 |
|  | { |
|  | class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | Console.Write("Enter equation: "); |
|  | string input = Console.ReadLine(); |
|  | int counter = 0; |
|  |  |
|  | for (int i = 0; i < input.Length; i++) |
|  | { |
|  | if (input[i] == '(') counter++; |
|  | if (input[i] == ')') counter--; |
|  | if (counter < 0) break; |
|  | } |
|  |  |
|  | if (counter == 0) Console.WriteLine("Correct equation."); |
|  | else Console.WriteLine("Wrong equation!"); |
|  |  |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that detects how many times a substring is contained in the text. For example, let’s look for the substring "**in**" in the text:

|  |
| --- |
| namespace Chapter\_13\_Solution\_5 |
|  | { |
|  | class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | int counter = 0; |
|  | string str = "We are living in a yellow submarine. We don't have anything else. Inside the submarine is very tight. So we are drinking all the day. We will move out of it in 5 days."; |
|  | Console.WriteLine(str); |
|  | str = str.ToLower(); |
|  | Console.Write("\nEnter substring to search: "); |
|  | string substr = Console.ReadLine().ToLower(); |
|  |  |
|  | int i = str.IndexOf(substr); |
|  |  |
|  | if (i != -1) |
|  | { |
|  | counter++; |
|  |  |
|  | while (i != -1 && counter >= 1) |
|  | { |
|  | i = str.IndexOf(substr, i + substr.Length); |
|  | counter++; |
|  | } |
|  |  |
|  | counter--; |
|  | } |
|  |  |
|  | Console.WriteLine("'{0}' is found {1} times in the text.", substr, counter); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. A text is given. Write a program that **modifies the casing** of letters to uppercase at all places in the text surrounded by **<upcase>** and **</upcase>** tags. Tags cannot be nested.

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_6 |
|  | { |
|  | public class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | string text = "We are living in a <upcase>yellow submarine</upcase>. We don't have <upcase>anything</upcase> else.", insideTag; |
|  | int startUpCase, endUpCase; |
|  |  |
|  | Console.WriteLine("Original text: \n{0}\n", text); |
|  |  |
|  | do |
|  | { |
|  | startUpCase = text.IndexOf("<upcase>", 0) + 8; |
|  | endUpCase = text.IndexOf("</upcase>", startUpCase); |
|  | insideTag = text.Substring(startUpCase, endUpCase - startUpCase).ToUpper(); |
|  | text = text.Remove(startUpCase, endUpCase - startUpCase); |
|  | text = text.Insert(startUpCase, insideTag); |
|  | text = text.Remove(startUpCase - 8, 8); |
|  | text = text.Remove(endUpCase - 8, 9); |
|  | } while (text.Contains("<upcase>") && text.Contains("</upcase>")); |
|  |  |
|  | Console.WriteLine("Modified text: \n{0}\n", text); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that reads a string from the console (20 characters maximum) and if shorter complements it right with "**\***" to 20 characters.

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_7 |
|  | { |
|  | public class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | Console.Write("Input text (20 char max): "); |
|  | string text = Console.ReadLine(); |
|  |  |
|  | text = text.PadRight(20, '\*'); |
|  |  |
|  | Console.WriteLine(text); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that converts a given string into the form of array of Unicode escape sequences in the format used in the C# language. Sample input: "**Test**". Result: "**\u0054\u0065\u0073\u0074**".

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_8 |
|  | { |
|  | public class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | Console.Write("Input text: "); |
|  | string text = Console.ReadLine(); |
|  | foreach (char c in text) Console.Write("\\u{0:x4}", ((int)c)); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that **encrypts a text** by applying XOR (excluding or) operation between the given source characters and given cipher code. The encryption should be done by applying XOR between the first letter of the text and the first letter of the code, the second letter of the text and the second letter of the code, etc. until the last letter of the code, then goes back to the first letter of the code and the next letter of the text. Print the result as a series of Unicode escape characters **\xxxx**.

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_9 |
|  | { |
|  | public class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | string text = "We are living in a yellow submarine. We don't have anything else. Inside the submarine is very tight. So we are drinking all the day. We will move out of it in 5 days."; |
|  |  |
|  | string[] sentences = text.Split('.'); |
|  |  |
|  | foreach (string str in sentences) |
|  | if (str.IndexOf(" in ") != -1 || str.IndexOf("In ") != -1) Console.WriteLine(str + '.'); |
|  |  |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that **extracts from a text all sentences that contain a particular word**. We accept that the sentences are separated from each other by the character "**.**" and the words are separated from one another by a character which is not a letter.

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_10 |
|  | { |
|  | public class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | string text = "Microsoft announced its next generation C# compiler today. It uses advanced parser and special optimizer for the Microsoft CLR.", forbiddenWords = "C#,CLR,Microsoft"; |
|  | string[] forbiddenWordsArr = forbiddenWords.Split(','); |
|  | string[] censoredForbiddenWordsArr = new string[forbiddenWordsArr.Length]; |
|  |  |
|  | for (int i = 0; i < forbiddenWordsArr.Length; i++) |
|  | censoredForbiddenWordsArr[i] = new string('\*', forbiddenWordsArr[i].Length); |
|  |  |
|  | for (int i = 0; i < forbiddenWordsArr.Length; i++) |
|  | text = text.Replace(forbiddenWordsArr[i], censoredForbiddenWordsArr[i]); |
|  |  |
|  | Console.WriteLine(text); |
|  | Console.ReadLine(); |
|  | } |
|  | } |

1. A string is given, composed of several **"forbidden" words** separated by commas. Also a text is given, containing those words. Write a program that **replaces the forbidden words with asterisks**. Sample text:

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_11 |
|  | { |
|  | public class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | Console.Write("Input number: "); |
|  | int number = System.Convert.ToInt32(Console.ReadLine()); |
|  |  |
|  | Console.WriteLine("{0, -15} {1, 15}", "Decimal:", number); |
|  | Console.WriteLine("{0, -15} {1, 15}", "Hexadecimal:", number.ToString("X")); |
|  | Console.WriteLine("{0, -15} {1, 15}", "Currency:", string.Format("{0:C}", number)); |
|  | Console.WriteLine("{0, -15} {1, 15}", "Percent:", string.Format("{0:P2}", number)); |
|  | Console.WriteLine("{0, -15} {1, 15}", "Scientific:", number.ToString("\\0.#####E0")); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |

1. Write a program that **parses an URL** in following format:

|  |
| --- |
| using System; |
|  |  |
|  | namespace Chapter\_13\_Solution\_15 |
|  | { |
|  | class Program |
|  | { |
|  | static void Main(string[] args) |
|  | { |
|  | string text = "<p>Please visit <a href=\"http://softuni.bg\">our site</a> to choose a software engineering training course. Also visit<a href=\"http://softuni.bg" + "/forum\">our forum</a> to discuss the courses.</ p >"; |
|  | text = text.Replace("<a href=\"", "[URL="); |
|  | text = text.Replace("\">", "]"); |
|  | text = text.Replace("</a>", "[/URL]"); |
|  |  |
|  | Console.WriteLine(text); |
|  | Console.ReadLine(); |
|  | } |
|  | } |
|  | } |