

Homework: Methods

This document defines the homework assignments from the ["Advanced C#" Course @ Software University](#). Please submit as homework a single **zip** / **rar** / **7z** archive holding the solutions (source code) of all below described problems.

Problem 1. Bigger Number

Write a method **GetMax()** with two parameters that returns the larger of two integers. Write a program that reads 2 integers from the console and prints the largest of them using the method **GetMax()**.

Sample Code	Input	Output
<pre>int firstNumber = int.Parse(Console.ReadLine()); int secondNumber = int.Parse(Console.ReadLine()); int max = GetMax(firstNumber, secondNumber); Console.WriteLine(max);</pre>	4 -5	4

Problem 2. Last Digit of Number

Write a method that returns the last digit of a given integer as an English word. Test the method with different input values. Ensure you name the method properly.

Sample Code	Output
<pre>Console.WriteLine(GetLastDigitAsWord(512)); Console.WriteLine(GetLastDigitAsWord(1024)); Console.WriteLine(GetLastDigitAsWord(12309));</pre>	two four nine

Problem 3. Larger Than Neighbours

Write a method that checks if the element at given position in a given array of integers is larger than its two **neighbours** (when such exist).

Sample Code	Output
<pre>int[] numbers = { 1, 3, 4, 5, 1, 0, 5 }; for (int i = 0; i < numbers.Length; i++) { Console.WriteLine(IsLargerThanNeighbours(numbers, i)); }</pre>	False False False True False False True

Problem 4. First Larger Than Neighbours

Write a method that returns the index of the **first element in array** that is **larger** than its **neighbours**, or **-1** if there's no such element. Use the method from the previous exercise in order to re.

Sample Code	Output
<pre>int[] sequenceOne = { 1, 3, 4, 5, 1, 0, 5 }; int[] sequenceTwo = { 1, 2, 3, 4, 5, 6, 6 }; int[] sequenceThree = { 1, 1, 1 }; Console.WriteLine(GetIndexOffFirstElementLargerThanNeighbours(sequenceOne)); Console.WriteLine(GetIndexOffFirstElementLargerThanNeighbours(sequenceTwo)); Console.WriteLine(GetIndexOffFirstElementLargerThanNeighbours(sequenceThree));</pre>	3 -1 -1

Problem 5. Reverse Number

Write a method that **reverses the digits** of a given **floating-point** number.

Sample Code	Input	Output
<pre>double reversed = GetReversedNumber(123.45); Console.WriteLine(reversed);</pre>	256	652
	123.45	54.321
	0.12	21

Problem 6. Number Calculations

Write methods to calculate the **minimum**, **maximum**, **average**, **sum** and **product** of a given set of numbers. **Overload** the methods to work with numbers of type **double** and **decimal**.

Note: Do not use LINQ.

Problem 7. * Generic Array Sort

Write a method which takes an array of any type and sorts it. Use **bubble sort** or **selection sort** (**your own implementation**). You may re-use your code from a previous homework and modify it.

Use a **generic method** (read in Internet about **generic methods in C#**). Make sure that what you're trying to sort can be sorted – your method should work with **numbers**, **strings**, **dates**, etc., but not necessarily with custom classes like Student.

Sample Code
<pre>int[] numbers = { 1, 3, 4, 5, 1, 0, 5 }; string[] strings = { "zaz", "cba", "baa", "azis" }; DateTime[] dates = { new DateTime(2002, 3, 1), new DateTime(2015, 5, 6), new DateTime(2014, 1, 1) }; SortArray(numbers); // 0, 1, 1, 3, 4, 5, 5 SortArray(strings); // azis, baa, cba, zaz SortArray(dates); // 3/1/2002 12:00:00 AM, 1/1/2014 12:00:00 AM, 5/6/2015 12:00:00 AM</pre>