Tutorial 4

Lecture revision

1. What language constructs does C# provides for loops?
2. What is difference between for, while, do while constructs?
3. How to exit a loop?

Programming exercise I– The Sum

What is the sum of all odd numbers between 1 and 1,000,000?

What is the sum of all even numbers between 1 and 1,000,000?

Write a program to answer these questions.

Programming exercise II– Interest calculator

Consider a bank account with n% interest rate per year. The interest is computed at the end of every year on the current balance and then deposited into the bank account. For example, if you deposit $10,000 initially and n = 5 (interest per year), after the first year you earned $500 (5% out of $10,000). The interest gets added to your bank account, making it equal to $10,500. The next year the interest is $525 (5% out of $10,500) and your balance becomes $11,025.

The task is to calculate total balance for given initial deposit, interest and number of years

1. Create new Windows application called InterestCalculator
2. Add controls to grab three input parameters, a button and a label to display the result
3. When user clicks the button you should grab the input into variables and iterate given number of times increasing balance by the interest rate specified
4. Display the result in the label

Programming exercise III– Prime input

Your goal is to write a program asking for an input from user and checking if the input is a prime number (which is only divisible over 1 and itself)

1. Create new Windows application called PrimeInput
2. Add a TextBox and a Button controls
3. Inside button’s click event handler check if input is numeric and if not – prompt for another input
4. If input is numeric you should check if the number entered is a prime number or not and show it in message box

Programming exercise IV – Show digits

Write a program that accepts an integer (via textbox) and breaks it into a sequence of individual digits in reverse order. For example, the input 16384 should be converted to

4

8

3

6

1

Hint: use % operator to get the last digit. Add it to a string variable to prepare the result. After that use integer division to get rid of the last digit of the number. Repeat until the number is out of digits.

Home work

Heron method

The Heron method is a method for computing square roots that was known to ancient Greeks. If *x* is a guess for, then the average of *x* and *a/x* is a better guess. Write a program that computes square root of a number. Stop when the difference between that number and x is less than 0.01 - where x is a square root of that number (Hint: start with initial guess equal to 1).

1. Create new Windows application called Heron
2. Add a NumericUpDown to form and a button
3. Declare a variable to hold your guess and set it to 1 by default
4. After button is hit enter a loop with the condition given (absolute value of difference between guessed number squared and actual number < 0.01) is not satisfied – use Math.Abs() function for getting absolute value
5. Within the loop reassign the variable with “better guess” – average of guess and input/guess
6. If the condition is satisfied – display both real square root (use Math.Sqrt() function to get it) and your guess via msgbox (use “\n” to separate lines)
7. Run the program and see the result