**Laboratory work 6**

**Part 1**

Create the web-page with HTML form, where include *Input* elements for entering and reading the value for the following exercises:

**Exercise 2**

Write a program to read Fahrenheit temperatures and print them in Celsius. The conversion formula is C = (5/9)(F - 32).

**Exercise 3**

Write a program that reads in the radius of a circle and prints the circle's diameter, circumference and area. Use the value 3.14159 for p.

**Exercise 5**

Get the user to input an integer number of seconds (i.e. 5049). Write a

program to convert this in hours, minutes and seconds. (The modulus operator will

come in handy). For example 5049 = H:1 M:24 S:9.

**Exercise 6**

A drinks machine manufacturer requires software for dispensing change.

One euro (100 cent) is the maximum that can be put into the machine. Given the

amount inserted and the cost of the item devise the change required. For example

suppose you put in 1 euro for an item costing 45 cent then your program should

generate output as follows:

Number of 50 cent coins : 1

Number of 20 cent coins : 0

Number of 10 cent coins : 0

Number of 5 cent coins : 1

Number of 2 cent coins : 0

Number of 1 cent coins : 0

**Part 2**

Exercise 1

Write four statements which each add 1 to an integer variable x.

Exercise 2 Write a program which invites the user to input 10 numbers and then

finally prints out the largest one entered.

Exercise 3 Write a program to print out all the Fibonacci numbers using short

integer variables until the numbers become too large to be stored in a short integer

variable i.e. until overflow occurs, (a) using a for loop construction and (b) using a

while loop construction. Which construction is most suitable in your opinion? Note:

Fibonacci numbers are (1, 1,2,3,5,8,13, etc.).

Exercise 4 Write a program which simulates the action of a simple calculator. The

program should take as input two integer numbers then a character which is one of +,

-, \*, /, %. The numbers should be then processed according to the operator input and

the result printed out. Your program should correctly intercept any possible erroneous

situations such as invalid operations, integer overflow, and division by zero.