CT240: Programming

- Date:
  - 13-September 2011
- Topic:
  - VB.NET
  - Lab structure
  - Lab for Thursday
  - Good Programming Practice
  - Input, Output and Expressions in VB.NET

What is required from labs:

- Problem Definition (taken from lab sheet) in your own words if possible
- Input-Process-Output Diagram
- Algorithm
- Program Code
- Test Results
- Screen shots of interface

Same “drill” as last year:

- “The tutors are there to help you to learn. Do not hesitate to ask them for support. Don't wait for them to get to you!
- You are expected to bring paper with you into the labs, and to figure out on paper how you will tackle a programming question **before** starting to write your code.
- You can hand up these hand-written notes for the IPO and Algorithm section of your solution; you don't have to type up everything. Whether you do it in hand-written form or typed, you should always work on your IPO and algorithm **before** writing your code.

- The tutors may ask to see your algorithm notes when you ask them for help with code.
- The tutors are not allowed to touch your keyboard!”

Why?!

- Paramount importance in programming is the *design* stage.
- Even if only working on small problems it is good programming practice to be forced to think of the problem first before coding
- Resulting code, even for smaller problems, will be clearer and stronger as a result of a separate design stage
- You will find it easier to describe your solution to lab tutors when seeking help

Why?

- Read: “The bug that destroyed a Rocket” by M. Ben-Ari
- “Students should ne encouraged to practice software development skills as early as possible. I hope that the story of Ariane 5 will help motivate them to do so”

Continuous Assessment Procedure this year:

- CT240 is examined by continuous assessment ONLY
- No written exam in December
- Your mark is made up of:
  - 7 Lab marks (best out of 8) + 1 = 30%
  - 2 Minor lab exams during term = 30%
  - 1 Major lab exam week after term = 40%
- The work done in assignments will lead directly in to the 2 tests.

Date	Task	Due	Worth
15-Sept	Assign 1	16-Sept	1
22-Sept	Assign 2	23-Sept	4
29-Sept	Assign 3	29-Sept	4
06-Oct	Test	end of lab	15
13-Oct	Assign 4	14-Oct	4
20-Oct	Assign 5	21-Oct	4
27-Oct	Assign 6	28-Oct	4
03-Nov	TEST	end of lab	15
10-Nov	Assign 7	11-Nov	4
17-Nov	Assign 8	18-Nov	4
24-Nov	Revision		
28-Nov**tbc	Test	end of lab	45

Final Test Monday 28-November

- Date and time needs to be confirmed
  - Will do so as quickly as possible
  - Hope to get time slot 11-1
- Will be based on ALL material covered during semester – labs and lectures
- Will be open book
- Best 6 of the 7 4-point assignments are chosen as assignment mark = 24% plus 1% for assignment 1 = 25%

Recall: Modular Programming Design

- Also applicable to event-driven programming
- Top-down design
- “Divide and conquer” approach
- Divide problem into logical tasks
  - Draw interface for tasks
  - Write code for each task
  - Test the whole

Assignment 1

- Task: Develop a program in VB.NET that gets your first name and surname and counts the number of letters in your first name and surname using the built in function *length()* in the string class.

Steps

- Draw IPO diagram
- Outline algorithm
- Draw GUI and set properties
- Write code
- Test

Required: `.length()` in VB.NET

- str.length()
  - Counts the number of characters in the string stored at *str*
  - Examples:

```
Dim str As String
str = “Hello there”
ans = str.length()
```

  - Note: Counts blank space.

What is the value of:
`ans = str.Length()`
if:

- str = “JG”
- str = “hello”
- str = “2”
- str = “1”
- str = “Room 405”
- str = “hello!”
- str = “hello!!!”

Input and Output

- For **output** in Visual Basic can use:
  - Label
  - Text Box
  - Message Box

- For **input** in Visual Basic can use:
  - Text Box
  - … and many others (menus, radio buttons, check boxes, files, etc.)

Using a Label

- Main purpose is to display descriptive text
- Can also be used for output but possibly not ideal

Example:

- Given a label, `lblTest` :

```
lblTest.Text = 3
lblTest.Text = "Here I am!"
```

Using a Text Box

- Allows users to enter, edit and output text
- The contents of a text box is always a string
- Numbers typed into text boxes are stored as strings and must be converted to numbers before being assigned to numeric variables
- Value of text boxes can be set to the empty string using “ ”

Example: Given a textbox, `txtTest`

For output:

```
Dim name As String
name = "Programming"
txtTest.Text = name
```

For input:

```
name = txtTest.Text
```

Using numbers and Text Boxes

- "The contents of a text box is always a string":
- What does this mean?
  - Text boxes always store things as strings … even if they look like numbers
  - Use `Str` to convert from a number to a string
  - Use `Val` to convert from a string to a number

Writing a number to a text box (`txtTest`) …
need to convert the number to a string

```
Dim age As Integer
age = 20
txtTest.Text = Str(age)
```

Getting a number from a text box (`txtTest`) ….
need to convert string to a number:

```
Dim ID As Integer
ID = Val(txtTest.Text)
```

What is an expression?

- Some meaningful combination of operators, variables, constants.
- Expressions inside parentheses are performed first.

Arithmetic Operators in VB.NET

+ - * / ^ MOD

- Examples: Give the values of the following variables after each assignment statement:

```
Private Sub btnTest_Click(ByV
    Dim x, y, z As Integer
    Dim a, b As Decimal

    x = 12
    a = 3.14
    y = x / 3

    z = x Mod 5

    b = a + 2.11

    x = 2 * x


End Sub
```

Evaluate each of the following:

```
Private Sub btnTest_Clic
    Dim x, n As Integer

    n = 10
    x = 0 Mod n
    x = 4 Mod n
    x = 12 Mod n
    x = 100 Mod n

End Sub
```

Find the value assigned to the integer variable a in each of the following assignment statements:

- a = 3 + 2 * 5 – 1
- a = a Mod 5
- a = a * a
- a = (3 + 2) * (5 – 1)
- a = ((3 + 2) * 5) – 1
- a = a Mod a

Example – Given a text box `txtAns` what will be displayed after each of the following lines

```
txtAns.Text = "Some Sums!"
txtAns.Text = 3 - 2
txtAns.Text = 3 * 2
txtAns.Text = 3 ^ 2
txtAns.Text = 2 * (3 + 4)
```

Example: Given a 2 textboxes, `txtNum1` and `txtNum2` and a command button, `btnTest` and a label, `lblOutput`:

```
lblOutput.text = txNum1.Text + txtNum2.Text
```

What is the output if:

3 is entered for Num1 and 5 entered for Num2?

2 is entered for Num1 and “day” is entered for Num2?

Summary

- Introduction to assignments and first assignment
- Introduction to VB.Net
- How we deal with Input and Output in VB
- Summary of Data:
  - What is a variable
  - What is a declaration
  - What is an assignment statement