Fundamentals of Computer Science
Exam – 10 September 2009

Name and Surname(s): _____

Previous notes:
a) Write your **name and surname**(s) on this sheet and immediately on all supplementary sheets, even drafts. Not doing it may suppose your expulsion.
b) You may use a **pencil** for your answers. You must turn your **mobile** off. You can neither use a **calculator**.
c) All students implied in the copy of an exercise will have a final mark of 0. Students are responsible to take care of their own exams.
d) "Receives" is different to "reads". "Returns" or "obtains" is different to "writes".
e) Use comments in the declarations to indicate which variable corresponds to which concept.

1. *(2 points)* As part of a program we want to show complex numbers without displaying two signs (+ –) when the imaginary part is negative.

a) Design the **flowchart** of a VB function receiving the real part (R) and the imaginary part (I) of a complex number and returns the string with the complex number, for example::

- R: -4,1; I: 6,23; **Result**: "-4,1 + 6,23i"
- R: 5,123; I: -9,2; **Result**: "5,123 - 9,2i"

b) **Implement** the VB program whose interface is shown in the following figure to read the real part and the imaginary part of a complex number (verifying that they are **numeric**) from two text boxes and then write the complex number utilising the function is section a).

Use the following **names** for the **controls**: txtR, txtI, cmdWr y pct1.
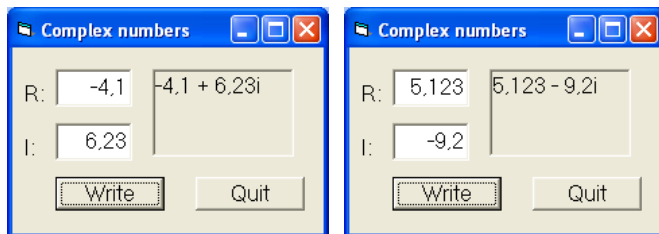
Reference table 1 of Visual Basic functions	
`IsNumeric(ByVal str As String) As Boolean`	Verify if the string *str* contains a numeric value
`Val(ByVal str As String) As Double`	Numeric value of a string *str*

2. *(2 points)* A function **S** is available that, given a positive integer index value **ind,** returns the value of the element in the position indicated by **ind** in a series of natural numbers s_1, s_2, ..., s_{ind}, ... which is always **increasing**. The function has the following header:
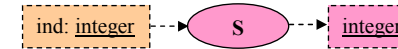
For example, given the following series:

$$S = \{s_1 = 3 , s_2 = 5, s_3 = 12, s_4 = 25, s_5 = 29, ...\}$$

- For an index **ind**=2, **S** returns: 5
- For an index **ind**=3, **S** returns: 12

Implement a VB function **IndIns** that receives a natural number **n** and returns the position of this number in the series given by **S** or a 0 if it does not belong to the series. For example, given the series in the previous example:

- for **n**=2 **IndIns** returns 0 (it is not part of the series)
- for **n**=3 **IndIns** returns 1 (it is in the position given by index 1)
- for **n**=4 **IndIns** returns 0 (it is not part of the series)
- for **n**=12 **IndIns** returns 3 (it is in the position given by index 3)

3. *(2.5 points)* **Implement** a VB function that, given a sentence and two characters (c1 and c2), returns if the following statement is true: "After the first appearance of the first character (c1) the second character (c2) appears at least three times".

Example:

- c1: "**n**", c2: "**e**", sentence: "No feeli**n**gs m**e**ans no soul and no r**e**al lif**e**"
  - Returned value: **True**

- c1: "**N**", c2: "**u**", sentence: "**N**o feelings means no so**u**l and no real life"
  - Returned value: **False**

Note: As it can be observed in these examples, characters c1 and c2 must match literally so that, for example, lowercase is different to uppercase.

Reference table 2 of Visual Basic functions	
`Mid(ByVal str As String, ByVal ini As Long, ByVal len As Long]) As String`	Substring from *ini* until the given length *len*, or until the end of the string if not specified
`Len (ByVal str As String) As Integer`	String length

4. *(2 points)* We have a digital watch that displays the current time through three data: hours, minutes and seconds (h, m and s, respectively). Its functionality consists in incrementing the previous time in one second and displaying it on the screen.

Examples:
- `h =  4, m =  5, s = 16.` **Result**: `h =  4, m =  5, s = 17`
- `h = 12, m = 53, s = 59.` **Result**: `h = 12, m = 54, s =  0`
- `h = 23, m = 59, s = 59.` **Result**: `h =  0, m =  0, s =  0`

a) **Implement** the `Increment1s` VB subprogram that receives one time (h, m and s), and modifies it adding one second.

b) **Implement** a VB program (associated to a control button) that asks the time (assume that it is correctly introduced) and displays on the screen the following time after incrementing it 1 second.

5. *(1.5 points)* **Implement** a VB function that receives a vector of real numbers and the number of significant elements within this vector and returns the sum of significant positive numbers.

Example:
- `v = {-1.3, 4.56, 2.5, 0, 2.897, -6.12}, n = 6`
- **Result**: `9.957`

Note: the vector may be assumed to be indexed starting from 1.