Escuela Universitaria
de Ingeniería
Vitoria-Gasteiz

Ingeniaritzako
Unibertsitate Eskola
Vitoria-Gasteiz

Fundamentals of Computer Science
2010-2011
Laboratory 9
Subprograms: Strings

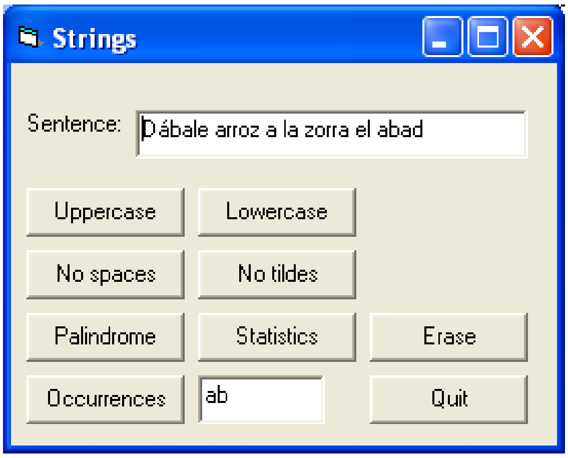
eman ta zabal zazu

Objectives:

- **Flowchart design** with **functions**
- **Implementation** of **functions** in Visual Basic
- Call to own and system functions
- **Character string** manipulation

Program to demonstrate the use of character strings

Interface



Sentence: Dábale arroz a la zorra el abad

ab

Figure 9.1 Objects present in the interface: command buttons, label and text boxes

Steps

1. Draw the whole **form**
2. Give a name to the text box containing the sentence, `txtSentence` and write the default text on it.
3. Give an appropriate name to the command buttons: `cmdUpp`, `cmdLow`, `cmdStat`, ...
4. Implement the "Quit" button to leave the application
5. Implement the "Erase" button to clear the sentence.
6. Implement the "Uppercase" and "Lowercase" buttons. Use **`Ucase`** and **`Lcase`** existing VB for it, consulting Table 9.1.
7. Implement the "No spaces" button to remove all the spaces in the sentence. Design for it the `RmSpaces` function that receives a string and returns a copy of it without spaces. Use **`Len`** and **`Mid`** VB functions consulting Table 9.1. The proposed algorithm is as follows:

    - Receive the original sentence in a parameter called `withS` (with spaces)

- Initialise a variable `withoutS` (without spaces) to the empty string
- Obtain the length `n` of the string `withS` by using the **`Len`** function (Table 9.1)
- Follow the whole string character by character and copy them from `withS` to `withoutS` whenever the character is not a space (" ").
  - To obtain the character `c` at position `i` of the string we make:

    ```
    c = Mid(withS, i, 1)
    ```

  - To "copy it" just concatenate this character to the `withoutS` string
- The function returns the contents of `withoutS`

8. Implement button "No tildes" to remove a characters with tildes (orthographic accents) from the sentence, substituting them with their equivalent without tilde. This way, "á", "à", "ä" and "â" will be substituted by "a" and similarly for the rest of the vowels. We shall also substitute from "ñ" for "n" and "ç" to "c".

   - Implement a function to convert one character:

     ```
     Function cnvTilde (ByVal c As String) As String
     ```

   - It's enough to find a solution with lowercase characters. If you want to find a smarter solution dealing with uppercase characters as well you can avoid duplicating the code you may obtain a copy of the original character, convert it to lowercase, deal only with lowercase characters and at the end of the function the resulting character to uppercase if the original was uppercase
9. Implement button "Palindrome" to say if a sentence is a pure palindrome. We call palindrome to a sentence that can equally be read in both senses like "bob", "madam", "Rats live on no evil star" or, in Spanish, "dábale arroz a la zorra el abad" (removing tildes and spaces). The algorithm is similar to the reverse in the previous laboratory, using characters instead of digits of a number.
10. Implement button "Statistics" to count characters, uppercase characters, lowercase characters, vowels and consonants. For each of these elements to count write a Boolean function that says if the given characteristic fulfils (uppercase, lowercase, etc.).
11. Implement button "Occurrences" to count how many times a string (or sub-string) in a text-box appears in the sentence.

Quick reference tables

`Mid (ByVal str As String, ByVal ini As Long, [ByVal len As Long]) As String`	Substring from *ini* until the given length, *len*, or the end of the string *str*
`Len (ByVal str As String) As Integer`	Length of *str* (number of characters)

`Ucase (ByVal str As String) As String`
`Lcase (ByVal str As String) As String`
	Convert *str* to lowercase of uppercase

Table 9.1 List of string functions in Visual Basic