Lab. 4 Text Processing; Input/Output to Text Files

Do the exercises below in the Octave IDE. Make sure the files and the programs are in the same working directory.

1. Text Processing

Create a sentence in a string variable, for example

`"This string (created for testing), has 70 characters, 17 being vowels."`

and use it to test the following functions that you should implement:

- `function c = n_chars(str)`
  – c the number of characters in the string str
- `function v = n_vowels(str)`
  – c the number of characters in the string str
- `function v = n_digits(str)`
  – c the number of characters in the string str
- `function v = n_words(str)`
  – c the number of characters in the string str
- `function v = n_integers(str)`
  – c the number of characters in the string str

2. Number of Substrings

a) Implement the following function, using no predefined MATLAB string functions

- `function n = n_occurs(sub, str, overlap)`

that returns the number of occurrences of the string `sub` in string `str`. `Overlap` is a Boolean parameter that allows or not starting a substring within another substring. For example given strings `str = "arara"` and `sub = "ara"` the function should return 2 if `over = True` but return only 1 if `over = False`.

b) Use predefined MATLAB string functions to implement an alternative version of the function.

3. Writing to a text File

a) Implement function below, that writes into the file with the specified `filename` all elements of vector `Vec`, in separate lines. The file should start with the sentence "`The following numbers are the k elements of a vector`" where **k** is the number of elements of the vector.

- `function write_vector(Vec, filename)`

b) Implement function below, similar to the previous one, but writing into the file with the specified filename all elements of matrix `Mat,` in separate lines, row by row. The file should start with the sentence "`The following numbers are the m * n elements of a matrix`" where m and n are, respectively the number of rows and columns of the matrix..

- `function write_matrix(Mat, filename)`

4. Reading from a text File

c) Implement functions below, that return, respectively, a vector and a matrix from files with name `filename,` with the format of those specified in the previous question.

- `function read_vector(filename)`
- `function read_matrix(filename)`