



INTERNATIONAL ADVANCED DIPLOMA IN COMPUTER STUDIES



MODULE: ADVANCED VISUAL BASIC ASSIGNMENT TITLE: Text Analysis SEPTEMBER 2009

Important Notes:

- ❖ Please refer to the Assignment Presentation Requirements for advice on how to set out your assignment. These can be found on the NCC Education *Campus*. Scroll down the left hand side of the screen until you reach Personal Support. Click on this, and then on Policies and Advice. You will find the Assignment Presentation Requirements under the Advice section.
- ❖ You must familiarise yourself with the NCC Education Academic Dishonesty and Plagiarism Policy and ensure that you acknowledge all the sources which you use in your work. The policy is available on *Campus*. Follow the instructions above, but click on Policies rather than Advice.
- ❖ You must complete the '**Statement and Confirmation of Own Work**'. The form is available on the Policies section of *Campus*. Scroll down the left hand side until you reach Personal Support. Click on this and then click on Policies and Advice.
- ❖ Please make a note of the recommended word count. You could lose marks if you write 10% more or less than this.
- ❖ You must submit a paper copy and digital copy (on disk or similarly acceptable medium). Media containing viruses, or media which cannot be run directly, will result in a fail grade being awarded for this module.
- ❖ All electronic media will be checked for plagiarism.

Marker's comments:

Moderator's comments:

Mark:

**Moderated
Mark:**

**Final
Mark:**

Introduction

The analysis of text using computer software was one of the early uses of computer applications. Initially, it was often used by academics to examine the authorship of written works. For example, to discover if the letters of St Paul in the Bible were all written by the same person.

Now text analysis is used extensively in commerce, entertainment, education, and politics. In fact, text analysis has become a pseudo science with its own terminology and base of mathematics. The process of text analysis starts with extracting a range of numerical data from blocks of text.

Note that: A block of text consists of a number of sentences.
 A sentence consists of a number of words.
 A word consists of syllables.

By extracting numerical information about these three entities (sentence, word, syllable) it is possible compare the readability of the text, the attitude of the author and perhaps identify the actual author.

For example, the underlying attitude of a political speech could be indicated by examining the relative use of the words:

- I
- We
- You
- They

This assignment is to design, implement and test software that will extract from text stored as a text file, the following information:

- Number of different words.
- Maximum sentence length.
- Minimum sentence length.
- Average sentence length.
- Lexical Density.*
- Gunning Fog Density.**

The software will also produce tables of the:

- Frequency of all words.
- Frequency of word length.
- Frequency of number of syllables in words.***
- Frequency of words that are entered by the user.

* Lexical Density is a measure of the readability of a piece of text, defined as:

$$\text{Lexical Density} = (\text{Number of different words} / \text{Total number of words}) \times 100$$

**Gunning Fog Density is a measure of the ease of understanding of a piece of text, defined as:

$$\text{Gunning Fog Index} = (\text{Average of number of words in sentence} + \text{percentage of words of three or more syllables}) \times 0.4$$

*** The complete algorithm for counting the number of syllables in a word is quite complex. However, it is acceptable in this assignment to obtain the number by counting the number of vowels in a word but ignoring adjacent same vowels e.g. as in the word 'good'.

Design and Implementation

You will want to test your system by using simple small blocks of text that you will first analyse by hand. Thus, it would be wise to have a text interface (separate form) that utilises a text box where the simple test data can be entered. This data will then be transferred to a text file that will be read by your Visual Basic algorithm.

Some of the results will be in tabular form, therefore consider the use of a grid screen object.

Blocks of text of reasonable lengths can be downloaded from the Internet. For example, pages from the Koran or the works of Shakespeare. Store these blocks as text files that can be read by your Visual Basic algorithms.

Ignore all punctuation except full stops, question marks and exclamation marks. (You have to have a means to detect the end of a sentence.) Ignore all capitalisation and all numeric characters. Make sure that a single space is a word delimiter, except at the end of a sentence.

Although the recommended textbook gives adequate coverage of the fundamentals of VB 2005 any particular assignment may require candidates to investigate other aspects of the language by using the Help facility provided by the Integrated Development Environment of VB 2005.

Aim

The aim of the assignment is to produce a simple text analysis software system as described above.

Task 1 – 5 Marks

Design the test interface as indicated in the ‘Design and Implementation’.

Task 2 – 10 Marks

Design some small blocks of test text. Produce hand analyses of these small blocks as indicated by the requirements of the system described in the introduction.

Implement and test the means to store these blocks as text files.

Task 3 – 10 Marks

Implement and test the design you produced in Task 1.

Task 4 – 20 Marks

Design all the forms necessary to enter user data and display the analysis required.

Include the design required to search the text file, generate the required numerical information and display the results.

Task 5 – 40 Marks

Implement and test the designs you produced in Task 4.

Show evidence of the validity of this testing with the help of screen shots.

Task 6 – 10 Marks

Obtain blocks of real text and store them as text files.

Use these files to live-run your completed system.

Produce evidence via screen shots of this live-running.

Task 7 – 5 Marks

Produce a publishable working copy of a compiled version of the completed assignment, together with some installation notes. The installation notes should include the system requirements.

This publishable copy, that includes a setup file, should be on an appropriate medium (CD-ROM, DVD etc.).

Guidance

The assessment of your project will to a large extent depend upon the quality of the documentation that you have produced. Thus, for **each stage** of the development of software:

- Give a detailed design including, where appropriate, the design of any algorithms.
- Build in error handling to involve meaningful messages that would ease any future maintenance of the software.
- Annotate all implementation.
- Design a testing strategy.
- Justify the design of suitable comprehensive test data.
- Show evidence of testing.
- Where appropriate, detail any major remedial action that you have taken in the light of the testing process.

Submission Requirements

A word-processed document incorporating the full documentation of each of the SEVEN Tasks. The document should be submitted both in paper form and digital form on a disk.

Refer to the Guidance above when producing your final documentation. Also refer to the Assignment Presentation Requirements on Campus. See front cover for where to find them.

Note the requirement in Task 7 to submit a publishable copy of the compiled system.

Warning: all media must be virus free!

Media containing viruses, or media which cannot be run directly, will result in a FAIL grade being awarded for this module.

You must read and understand NCC Education's policy on 'Academic Dishonesty and Plagiarism'.

You must complete the 'Statement and Confirmation of Own Work' form and attach the completed form to your assignment.