UNIVERSITY OF SWAZILAND

Faculty of Science

Department of Computer Science

SUPPLEMENTARY EXAMINATION 2009

Titles of paper: **SOFTWARE ENGINEERING – I** and
                     SOFTWARE ENGINEERING FOR B.ENG.

Course numbers: **CS451** and **CS461**

Time allowed: 3 hours

Instructions: Answer question 1 and <u>any 4</u> of the remaining 5 questions. Each question carries 20 marks.

Question 1 (Compulsory)

Carefully read the following information and answer questions a) and b) further below

A spelling checker is required to provide 5 principal features, as follows:
1 INPUT TEXT FOR CHECKING
    1.1   The software inputs text for spell-checking from any text file specified by the user.
    1.2   Alternatively, the user may type a word directly into the spelling checker.

2 SPELL CHECKING
    2.1   The software checks spelling of text in any language for which a dictionary is available. A dictionary can be considered to be a text file containing all valid words in a language. The dictionary file contains one word per line.
    2.2   The user selects the relevant dictionary at the commencement of spell checking.

3 INPUT OF CORRECTIONS
    3.1   The software suggests corrections for misspelled words by searching the dictionary for words resembling the misspelled one.
    3.2   Allows the user to replace misspelled words with one of the suggested corrections.
    3.3   When informed of a misspelling, the user is allowed to reject the suggested corrections and type a different word of his choice, or even ignore the misspelling altogether.

4 DICTIONARY MODIFICATION
    4.1   If the user types an unrecognized word to correct a misspelling, the software asks the user whether the new word is to be added to the dictionary.
    4.2   Likewise, if the user opts not to correct a (supposed) misspelling, the software asks the user whether the word is to be added.

5 SAVING
    5.1   If text was input from a file (i.e. not typed into the spelling checker), then at the end of spell checking, the software allows the user to save the file if desired.

Questions
    a) Formulate a design by Functional Decomposition for the software. Your answer should be presented as a hierarchy diagram of functions. In addition, write a sentence on the purpose of each function.   [10]

    b) Present a Data Flow design in the form of a Data Flow Diagram (DFD) for the software. In addition, write a sentence on the purpose of each process in the DFD.
                 [10]

Question 2

a) Distinguish between verification and validation. [2]

b) Define any 4 principles of software engineering ethics laid out in the IEEE/ACM code of ethics. For each principle you mention, describe a realistic professional scenario in which the principle would apply. [12]

c) Explain the statement: "Managers can exert control over the following aspects of a project: time, information, organization, quality and money." [6]

Question 3

a) What is meant by *critical path* in project planning? [2]

b) Describe the Waterfall lifecycle model and explain its main shortcomings. [12]

c) Describe the Spiral lifecycle model. [6]

Question 4

a) Distinguish between requirements elicitation and requirements specification. [2]

b) Describe any 4 requirements elicitation techniques. [8]

c) List the contents of the "Specific Requirements" section of the IEEE standard for requirements specifications (IEEE 830-1993 section 3). In addition, briefly describe the meaning of each listed item. [10]

Question 5

a) Define the terms *module, cohesion* and *coupling* as applied to software design. [8]

b) Define any 4 kinds of *coupling*, listed in order of increasing strength. [6]

c) Define any 4 kinds of *cohesion*, listed in order of increasing strength. [6]

Question 6

a) Distinguish between the following sets of terms:
   i. Error, fault and failure
   ii. Black box and white box testing
   iii. Retest-all and selective retest [6]

b) Define the following kinds of software testing: coverage-based, fault-based and error-based. [6]

c) Define the following kinds of maintenance: corrective, adaptive, perfective and preventive. [8]