PRINCE OF SONGKLA UNIVERSITY
FACULTY OF ENGINEERING
Department of Computer Engineering

Exam Duration: 2 hours

This paper has 2 pages.

Authorized Materials:

- Writing instruments (e.g. pens, pencils).
- Books (e.g. dictionaries) and calculators are **not** permitted.

Instructions to Students:

- *Answer questions in English.* Perfect English is **not** required.
- Attempt all questions.
- Write your answers in an answer book.
- Start your answer to each question on a new page
- Clearly number your answers.
- Any unreadable parts will be considered wrong.
- When writing programs, use good layout, and short comments; marks will not be deducted for minor syntax errors.
- The marks for each part of a question are given in brackets (...).

Question 1   (15 minutes; 15 marks)

a) Write a regular expression using grep (or egrep) which lists all the words in /usr/share/dict/words that use some or all of the letters q, w, e, r, t, y (e.g. tree, wetter). (5)

b) Write a regular expression using grep (or egrep) which lists all the words in /usr/share/dict/words that contain at least two e's and no other vowels (e.g. wheezes). The vowels are a, e, i, o, and u. (10)

Note: your answers may use multiple calls to grep/egrep piped together.

Question 2   (20 minutes; 20 marks)

Write a **lex** program which finds all the floats in an input file, and prints their sum when the end of file is reached. *Note*: a float is a number with a decimal point (e.g. 12.456, 5.0, 0.001). *Note*: use atof() to convert a char array into a float.

Question 3   (45 minutes; 45 marks)

a) Explain the four parts of a grammar. Do **not** include any program code. (10)

b) Write a context-free grammar that generates a sequence of *a*'s followed by a sequence of *b*'s, where there is *one more* b than a (e.g. abb, aabbb).

Draw a parse tree for the string aabbb using your grammar. (10)

c) Write a context-free grammar for expressions involving integers, *, and / (e.g. 14*5/22, 3*12*114).

Draw a parse tree for the string 3*12*114 using your grammar. (15).

d) Explain grammar *ambiguity*, and include an example. (10)

Question 4   (40 minutes; 40 marks)

a) What are FIRST sets? Define FIRST() using FIRST_SEQ(). (10)

b) What are FOLLOW sets? Define FOLLOW() using FIRST_SEQ(). (10)

c) Calculate the FIRST **and** FOLLOW sets for the grammar:

S → (A) | ε

A → B C

B → (A) | a | b | c

C → @ B C | ε

The lowercase letters, (,), and @ are terminals. The uppercase letters are non-terminals, and S is the start symbol. Show **all** your working. (20)

--- End of Examination ---