PRINCE OF SONGKLA UNIVERSITY
FACULTY OF ENGINEERING
Department of Computer Engineering

Midterm Examination: Semester 1

Date: 3rd August, 2010

Subject Number: 241-437

Subject Title: Compiler Structures

Lecturer: Aj. Andrew Davison

Academic Year: 2010-2011

Time: 9:00 – **11:00 (2 hours)**

Room: A401

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   (20 minutes; 20 marks)

a) What is a lexical analyzer? Explain with the help of diagrams, but do **not** include any code examples. (5)

b) Write a lex program which counts the number of words contained in the input file. (10)

c) Explain how the program in part (b) works, especially how a word is defined, and how lex matches the program's rules against the input. (5)

Question 2   (20 minutes; 20 marks)

a) Explain the four parts of a grammar. Give brief examples, but do **not** include any program code. (10)

b) Write a grammar for expressions which use the '+' and '*' operators, brackets, and whose operands are integers. Example expressions include "23+5" and "(5*2)+67". (10)

Question 3   (30 minutes; 30 marks)

a) What are FIRST sets? Define FIRST() using FIRST_SEQ(). (8)

b) What are FOLLOW sets? Define FOLLOW() using FIRST_SEQ(). (8)

c) Calculate the FIRST and FOLLOW sets for the expressions grammar that you wrote in Question 2(b). (14)

Question 4   (50 minutes; 50 marks)

a) What is a LL(1) grammar? Give an example, but do **not** include any program code. (10)

b) What are PREDICT sets? Define PREDICT() using FIRST() and FOLLOW(). Do not define FIRST() and FOLLOW(). (10)

c) Use PREDICT() to check the expressions grammar that you wrote in Question 2(b). (15)

d) Explain how a recursive descent parser is generated by translating your expressions grammar into C functions. Include code for main(), but **not** for nextToken(), match(), error(), or scanner(). (15)

--- End of Examination ---