MASENO UNIVERSITY

UNIVERSITY EXAMINATIONS 2016/2017

FOURTH YEAR FIRST SEMESTER EXAMINATION FOR DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

MAIN CAMPUS

SCS 401: COMPILER CONSTRUCTION

Date: 13th December, 2016 Time: 8.30 - 11.30am

INSTRUCTIONS:

- Answer ALL Questions in Section A and any other TWO in Section B.
- Write your registration number on all sheets of the answer book used
- Use a new page for every Question attempted, and indicate the question number on the space provided on each page of the answer sheet
- Fasten together all loose answer sheets used
- No mobile phones in the examination room.

SECTION A: ANSWER ALL QUESTIONS

Question One (30 marks)

a) State and briefly explain four major types of data structures in a compiler. **(4 marks)**

b) Backus-Naur Form (BNF) for simple arithmetic expressions may be represented as:

< exp > → < exp > + < exp > | < exp > − < exp > |< exp > * < exp > | < exp > / < exp > |(< exp >) | < number >

< number > → < digit > | < digit > < number >

< digit > → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9.

Write a leftmost derivation of 8 + 6 – 4 * 7/2 then draw its equivalent parse tree. **(6 marks)**

c) State and briefly explain five types of errors that can be encountered during the compilation phases. **(5 marks)**

d) Use appropriate symbolic notation to explain the components of a context-free grammar. **(5 marks)**

e) Explain the purpose of a symbol table in the compilation process, stating typical attributes that can be defined for variables, procedures and functions. **(5 marks)**

f) Compare compilers and interpreters as programming language translation mechanisms. **(5 marks)**

SECTION B: ANSWER ANY TWO QUESTIONS

Question Two (20 marks)

a) Discuss code optimization within a compiler, clearly explaining the main categories of optimization and some specific areas that have been known to cause performance measurements problems of compilers. **(4 marks)**

b) Use appropriate examples to explain the various techniques that constitute the following target code optimizations

i) Loop optimizations **(8 marks)**

ii) Function-preserving transformations **(8 marks)**

Question Three (20 marks)

The grammar below is an LL(1) grammar for regular expressions over alphabet {a, b}, with + standing for the union operator and Є for the symbol ε.

E →TE'

E' →+E | ε

T →FT'

T' →T | ε

F →PF'

F' →*F' | ε

P →(E) | a| b| Є

a) Compute FIRST and FOLLOW for each nonterminal of the above grammar. **(6 marks)**

b) Show that the grammar is LL(1). **(2 marks)**

c) Construct the predictive parsing table for the grammar. **(6 marks)**

d) Determine whether the string b*a(a+b) can be parsed by the grammar. **(6 marks)**

Question Four (20 marks)

a) Use simple diagrams to distinguish between deterministic finite state automaton (DFA) and non deterministic finite state automaton (NFA) **(6 marks)**

b) Build a scanner represented by the following regular expressions using DFA **(5 marks)**

while-keyword = while

Identifier = [a- z][a- z0 - 9]*

c) Develop a state transition table for the above scanner **(5 marks)**

d) Given the grammar G with the following production rules, S →a | aS | bS, represent the language denoted by the grammar in regular expression, then determine whether the following strings can be generated by the grammar (i) babaaba (ii) aaabbbab (iii) bbaaba **(4 marks)**

Question Five (20 marks)

Describe the different phases of a compiler by clearly using relevant examples to explain the activities taking place in each phase.

(20 marks)