CHUKA  **UNIVERSITY**

UNIVERSITY EXAMINATIONS

EXAMINATION FOR THE AWARD OF DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

COSC 327: COMPILER CONSTRUCTION

STREAMS: BSC (COMP. SCI) Y3S2 **TIME: 2 HOURS**

DAY/DATE: MONDAY 08/4/2019 **2.30 P.M. – 4.30 P.M.**

INSTRUCTIONS: Answer Question One and any other two questions

Question One (Compulsory) (30 marks)

(a) Explain the difference between a compiler and an interpreter. (4 marks)

(b) Explain when a grammar G is said to be ambiguous. Give an example to illustrate your answer together with a demonstration that it's indeed ambiguous. (4 marks)

(c) At a given point in the execution of a program, what can be considered as garbage? How can garbage be located in memory? (4 marks)

(d) Explain two reasons why stack is used for storing procedure calls in compiler construction. (4 marks)

(e) Using a C sample code illustration, explain the difference between formal parameters and actual parameters in functions. (4 marks)

(f) Identify four reasons of creating intermediate code during compilation as opposed to directly translating source code to machine code. (4 marks)

(g) Discuss the three rules of code optimization. (6 marks)

Question Two (20 marks)

(a) State and explain the two major operations on a symbol table. (4 marks)

(b) Consider an optimization that replaces any expression of the form

$0 \times e$

with 0, where e is an arbitrary expression. Why might this rule be useful in a compiler's optimization phase? Is it always correct? (6 marks)

(c) Let *G* be the following grammar:

S -> A
A ->B | B a A
B -> b C
C -> Cb | Cc | ε

Modify *G* to produce a new grammar, *G'*. Show that *G'* is LL(1). Describe the condition(s) that *G'* meets that makes it LL(1). (10 marks)

Question Three (20 marks)

(a) Explain what a symbol table is and state its importance in compilation process.(4 marks)

(b) Describe four optimizations likely to be performed on a piece of code during compilation. (8 marks)

(d) Consider the following grammar:

```
S  →   ( L )
       | a
L  →   L , S
       | S
```

Under this grammar, derive the parse tree of the sentence (a, ((a, a), a)) (8 marks)

Question Four (20 marks)

(a) Given production rules, generate a parse tree for the input string:

```
b = a * x / y + z * x/y;
```

Use left most derivation. Show all the steps involved. (10 marks)

The production rules:

E → E + E
E → E / E
E → E * E
E → id
E → num

(c) For each of the production rules (c) above, write their corresponding semantic rules. (4 marks)

(d) Generate a Directed Acyclic Graph from the syntax tree in (a) above. (6 marks)

Question Five (20 marks)

(a) Outline four semantic errors that a compiler should check and report. (4 marks)

(b) Explain the following compilation stages. (8 marks)

(i) Syntax analysis
(ii) semantic analysis
(iii) code generation
(iv) code optimization

(c) Three address code and instructions for the instructions shown below and represent them in a quadruple. (8 marks)

```
y = z * x
while (y > z){
     x = y * z
     y = z – x
     z = y * z
}
```
