Strathmore
UNIVERSITY

FACULTY OF INFORMATION TECHNOLOGY
BACHELOR OF INFORMATICS & COMPUTER SCIENCE
END OF SEMESTER EXAMINATION

ICS 4211: COMPILER CONSTRUCTION/DESIGN

MAIN EXAM

DATE: **Time:**

Instructions

1. This examination consists of **FIVE** questions.
2. Answer **Question ONE (COMPULSORY)** and any other **TWO** questions.

QUESTION ONE **(Total: 20 Marks)**

a. What is the difference between the parser and semantic analyser? (2 Marks)

b. Write a simple C++ or C code that generate tokens from a given string. Highlight how your code works using a sample string of your own. (4 Marks)

c. Regular expression questions:

i. Write a regular expression for all strings of ***as*** and ***bs*** which contains the substring ***abba***

(3 Marks)

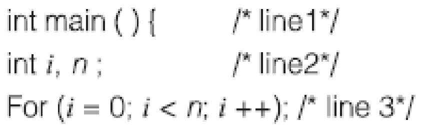
ii. Write a regular expression for all strings of ***xs*** and ***ys*** where every ***y*** is immediately followed by at least 3 ***xs*** (3 Marks)

d. Using your vast knowledge of compiler design, why do you think the front end and back end are separated in compiler design?

(3 Marks)

e. What are different kinds of errors encountered during compilation? (3 Marks)

f. Consider line 3 in the C-program below.



```
int main ( ) {          /* line1*/
int i, n ;              /* line2*/
For (i = 0; i < n; i ++); /* line 3*/
```

Identify the compiler's response about line 3 when creating the object module.

(2 Marks)

QUESTION TWO **(Total: 20 Marks)**

a. Compilers translate code from human readable form into machine code, a highly complex problem owing to the difference between source language and generated code. To manage the complexities, the translation is done in passes. Differentiate the different types of compilers. (Hint: Illustrations are highly welcomed). (5 Marks)

b. Translate the conditional statement *if a<b then 1 else 0* into three address code. (4 Marks)

c. Give a regular expression for the intersection, union, and concatenation respectively of the two languages: A = $\{w \in \{0,1\}^*$: w begins with 11} and B =$\{w \in \{0,1\}^*$: w ends with 00} (6 Marks)

d. What are the solutions to the top down parsing problems? (3 Marks)

e. Name any two phases of a compiler and describe their functions. (2 Marks)

QUESTION THREE **(Total: 20 Marks)**

a. Given the following grammar:

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T \times F \mid F$$

$$F \rightarrow (E) \mid id$$

Generate the following for the string "*id + id x id*"

i. Parse tree (4 Marks)

ii. Syntax tree (2 Marks)

iii. Directed Acyclic Graph (DAG) (1 Mark)

b. Consider the following context-free grammar for the following questions

$$\begin{aligned} S &\rightarrow A\$ \\ A &\rightarrow (AB \\ A &\rightarrow \lambda \\ B &\rightarrow (A) \\ B &\rightarrow x \quad : \end{aligned}$$

i. What are the terminals and non-terminals of this grammar? (3 Marks)

ii. Show the parse tree for "((x)x)$" (4 Marks)

iii. Give the first and follow sets for each of the non-terminals of the grammar. (3 Marks)

c. In the context of complier construction, what is an ambiguous grammar and how would you deal with it? (3 Marks)

QUESTION FOUR **(Total: 20 Marks)**

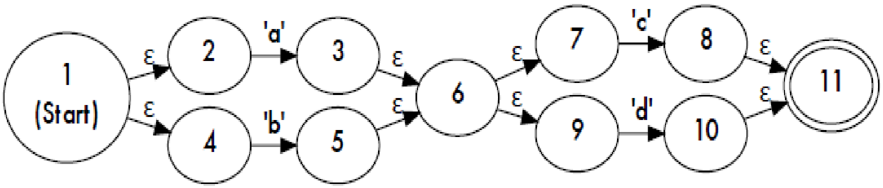
a. Describe the concept of derivation and the two most important elements of this concept. (3 Marks)

b. Write the algorithm for FIRST and FOLLOW. (5 Marks)

c. Explain the usage of YACC parser generator in construction of a Parser. (4 Marks)

d. Name an advantage and disadvantage of separate compilation. (2 Marks)

e. Convert the NFA below into its equivalent DFA using the "sets of states" construction process. Indicate the set of states that each DFA state corresponds to.
(6 Marks)



QUESTION FIVE **(Total: 20 Marks)**

a. Describe the code generation module and the issue it must tackle when executing its roles. (5 marks)

b. What is the "output" of the below pseudo code assuming bindings are resolved using the closest nested scope rule and (respectively)

   i. dynamic scoping; and (2 Marks)

   ii. Lexical (i.e., static) scoping? (2 Marks)

```
VAR count: INTEGER;

PROCEDURE procX IS
BEGIN
  VAR count: INTEGER;
  SET count TO 100;
  CALL report
END

PROCEDURE procY IS
BEGIN
  SET count TO 200;
  CALL report
END

PROCEDURE report IS
BEGIN
  PRINT "count = " + count
END

MAIN PROGRAM IS
BEGIN
  SET count TO 300;
  CALL procX;
  CALL procY
END
```

c. What are the benefits of intermediate code generation? (4 Marks)

d. Using the common intermediate representations of programs, write the Three Address Code of the following program lines

   i. X = (y + z) *w (2 Marks)

   ii. If x <= 0 Then
           y = z
       End If (2 Marks)

   iii. While a < b

```
    If c < d Then
        x = (y + z) * w
    Else
        x = (y – z) * w
    End If
End While
```

(3 Marks)