MACHAKOS UNIVERSITY

University Examinations for 2022/2023

SCHOOL OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTING AND INFORMATION TECHNOLOGY

THIRD YEAR SECOND SEMESTER EXAMINATION FOR

BACHELOR OF SCIENCE (COMPUTER SCIENCE)

SCO301: COMPILER CONSTRUCTION

DATE: **TIME:**

INSTRUCTIONS: Answer Question ONE and Any Other TWO Questions.

QUESTION ONE (COMPULSORY) (30 MARKS)

a) Using a simple illustration describe the overview of language processing system. (4 marks)

b) Compiler optimizations must meet various design objectives. Explain FOUR of such design objectives. (4 marks)

c) Analyze FOUR reasons for using the regular expression to define the lexical syntax of a l anguage. (4 marks)

d) In compiler construction an activation record contains all the necessary information required to call a procedure. Explain types of that can be used depending on the source language used. (4 marks)

e) The most important criterion for a code generator is that it produces a correct code. However, there are issues that do arise during the code generation phase. Explain FIVE of these common issues. (5 marks)

f) List FIVE characteristics of peephole optimization in compiler construction. (5 marks)

g) Contrast between Right Recursion and Left Recursion. (4 marks)

QUESTION TWO (20 MARKS)

a) Distinguish the between the TWO main phases of compilation. (4 marks)

b) Describe the TWO main buffering techniques employed in a compiler. (4 marks)

c) The lexical- (scanner), syntactic- (parser), and semantic-analysis phases of a compiler front-end each process parts of the source program in particular ways and also check certain rules of the language being compiled. For each of the following possible language rules, specify which phase of the compiler should verify that a program conforms to that rule and why that part of the compiler is the best place for that check. If a check could be done equally well in more than one phase of the compiler, briefly discuss the tradeoffs between the alternative implementations.

i. A function is called with the correct number of arguments. (3 marks)

ii. Underscore characters (_) may appear in the middle of identifiers, but not at the beginning or end (i.e., this_identifier is legal, but _this_one is not). (3 marks)

iii. Every variable must be declared before it is used in the program (the classic C or Pascal rule). (3 marks)

iv. Assignment statements must end with a semicolon (;). (3 marks)

QUESTION THREE (20 MARKS)

a) Compiler design impacts several other areas of computer science. Explain FOUR of such areas in the computer science field. (4 marks)

b) Describe FOUR error-recovery actions that can be taken when a parsing or lexical error is detected. (4 marks)

c) Illustrate the LR parsing algorithm as used in compiler constructions. (4 marks)

d) Consider the context-free grammar

$$S \rightarrow S\,S+ \mid S\,S* \mid \mathbf{a}$$

i. Show how the string aa+a* can be generated by this grammar. (4 marks)

ii. Construct a parse tree for this string. (4 marks)

QUESTION FOUR (20 MARKS)

a) Discuss THREE important conventions about the transition diagrams. (6 marks)

b) Analyze FOUR advantages of LR parsing. (4 marks)

c) Let G be the grammar:

$$S \rightarrow A\$$$
$$A \rightarrow (AB)$$
$$A \rightarrow \lambda$$
$$B \rightarrow (A)$$
$$B \rightarrow x$$

Using this grammar, describe the terminals and non-terminals of the above grammar.

(4 marks)

d) Draw a parse tree for the string S = "((x)x)$" (6 marks)

QUESTION FIVE (20 MARKS)

a) Compare the TWO types of parsing. (4 marks)

b) Discuss TWO roles of the lexical analyzer in the phases of the compiler. (6 marks)

c) Discuss TWO methods for improving a program in compiler construction. (4 marks)

d) Summarize on how context-free grammars are specified in compiler construction.

(6 marks)