



UNIVERSITY EXAMINATION 2020/2021

YEAR 3 SEMESTER II EXAMINATION FOR THE DEGREE OF BACHELOR OF
TECHNOLOGY IN ELECTRONICS & COMPUTER ENGINEERING.

EEE 2551 Compiler Construction Year 3 Semester II

Date: Friday 17th September, 2021

Time: 1.30 pm – 3.30 pm

Instructions: Answer Question 1 and Any Other Two.

Show all your working to earn all the marks

Where possible illustrate with diagrams

Question One: (30 Marks)

- a) Distinguish tokens, patterns and lexeme [4 marks]
- b) Explain two difference between La-attributed and S- attributes in syntax
directed translation [4 marks]
- c) Give at least one error produced by each stage of the Front End of a
compiler [4 marks]
- d) Consider the grammar where S is non-terminal:

$$\begin{array}{l} S \rightarrow SS + \\ \quad | SS * \\ \quad | a \end{array}$$

Given the input $a a + a *$ show the difference between parse and syntax

tree by use of graphs [4 marks]

- e) Construct Three address code for the following quadratic equation below
$$y := a^2 - ab + c^2$$
 [4 marks]
- f) Consider the following error cases which occurred during operator parsing
and state the recovery strategy: No relation hold between the terminal on
the top of stack and the next input and a handle is found (reduce step), but
there is no production with this handle as a right side [4 marks]
- g) Given the Syntax-Directed Definition below construct the annotated parse
tree for the input expression: "int a, b, c" [4 marks]

$$\begin{array}{ll} D \rightarrow T L & L.inh = T.type \\ T \rightarrow int & T.type = integer \end{array}$$

h) What are the various ways to pass a parameter in a function?[2 marks]

Question Two: (20 Marks)

a. Consider the following grammar:

$$\begin{array}{l} E \rightarrow TE' \\ E' \rightarrow +TE' \mid \epsilon \\ T \rightarrow FT \\ T' \rightarrow *FT' \mid \epsilon \\ F \rightarrow (E) \mid id \end{array}$$

i. Compute the First and Follow sets for the non-terminals in this grammar. [10 marks]

ii. Construct the parse table for predictive parser for this grammar [10 marks]

Question Three: (20 Marks)

a. Give the three rules of code optimization in compiler construction [3 marks]

b. Consider the program below and translate to its basic building blocks [4 marks]

```
w = 0;
x = x + y;
y = 0;
if( x > z)
{
    y = x;
    x++;
}
else
{
    y = z;
    z++;
}
w = x + z;
x = x - y;
```

c. Consider the code below and create the relevant intermediate code while

```
while (x < y + z) {
    x = x - y;
}
```

[6 marks]

d. Give a post fix notation intermediate code for the following

[7 marks]

```

If a+b then
  If c-d then
    x := 10
  else
    y := 20
else
  z := 30;

```

Question Four: (20 Marks)

- Explain what is made by type checking [2 marks]
- Susan is designing a Kamba compiler, As an expert she comes to you for advice on the components she should have on a her type checker, explain any three [6 marks]
- Consider the syntax directed grammar below for evaluating signed binary number to decimal, given -101 as the input by drawing a parse tree evaluate the node until you reach the root and state the output [12 marks]

PRODUCTION	SEMANTIC RULES
NUM \rightarrow SIGN LIST	LIST.pos := 0 if SIGN.neg NUM.val := -LIST.val else NUM.val := LIST.val
SIGN \rightarrow +	SIGN.neg := false
SIGN \rightarrow -	SIGN.neg := true
LIST \rightarrow BIT	BIT.pos := LIST.pos LIST.val := BIT.val
LIST \rightarrow LIST ₁ BIT	LIST ₁ .pos := LIST.pos + 1 BIT.pos := LIST.pos LIST.val := LIST ₁ .val + BIT.val
BIT \rightarrow 0	BIT.val := 0
BIT \rightarrow 1	BIT.val := $2^{\text{BIT.pos}}$

Question Five: (20 Marks)

Consider the grammar below and construct the action goto table show all your working and explain any problem encountered [20 marks]

- $E \rightarrow E+T$
- $E \rightarrow T$
- $T \rightarrow T \cdot F$
- $T \rightarrow F$
- $F \rightarrow (E)$
- $F \rightarrow \text{id}$