USN | | | | | | | | | | |

06CS63

Sixth Semester B.E. Degree Examination, June/July 2019

Compiler Design

Time: 3 hrs. Max. Marks:100

Note: ***Answer any FIVE full questions, selecting at least TWO questions from each part.***

PART – A

1 a. Explain with a neat diagram the phases of compiler. **(08 Marks)**

b. Write a note on language processors. **(06 Marks)**

c. Explain the role of lexical analyzer. **(06 Marks)**

2 a. What is left recursion? Write an algorithm to eliminate left recursion. **(06 Marks)**

b. Construct predictive parsing table for the grammar.

$E \rightarrow TE'$

$E' \rightarrow +TE'/\in$

$T \rightarrow FT'$

$T' \rightarrow *FT'/\in$

$F \rightarrow (E)/id$ **(14 Marks)**

3 a. Explain shift-reduce parsing actions. **(04 Marks)**

b. Consider the Augmented expression grammar

$E' \rightarrow E$

$E \rightarrow E+T/T$

$T' \rightarrow T*F/F$

$E \rightarrow (E)/id$

Obtain LR(0) automation. **(10 Marks)**

c. Write construction of SLR-parsing table algorithm. **(06 Marks)**

4 a. Consider the grammar $S' \rightarrow S$, $S \rightarrow CC$, $C \rightarrow cC/d$. Construct GOTO graph for the grammar. **(10 Marks)**

b. Obtain canonical parsing table for the grammar given in the question.4(a). **(05 Marks)**

c. Write LALR parsing table construction algorithm. **(05 Marks)**

PART – B

5 a. Write the syntax-directed definition of (SDD) simple desk calculator:
$L \rightarrow En$, $E \rightarrow E + T$, $E \rightarrow T$, $T \rightarrow T*F$, $T \rightarrow F$, $F \rightarrow (E)$, $F \rightarrow$ digit **(06 Marks)**

b. Draw an annotated parse tree for the expression $3*5 + 4n$ using SDD obtained in question.5(a). **(06 Marks)**

c. Consider the grammar $E \rightarrow E + T$, $E \rightarrow E - T$, $E \rightarrow T$, $T \rightarrow (E)$, $T \rightarrow$ id, $T \rightarrow$ num. Write and explain S - attributed definition to construct syntax tree. **(08 Marks)**

6 a. Discuss the value number method for constructing DAG for the example $i = i + 10$. **(04 Marks)**

b. What is three address code? Write down a DAG and a three address code for the expression $a + a * (b - c) + (b - c) * d$. **(06 Marks)**

c. Write the syntax directed definitions for flow of control statements and explain. **(10 Marks)**

7 a. Explain activation records with neat diagram. **(10 Marks)**

b. Discuss reference counting garbage collection with an example. **(10 Marks)**

8 a. Describe the various issues in the design of code generator. **(10 Marks)**

b. Write notes on the following with example:

i) Dead code elimination

ii) Local common sub expression **(10 Marks)**

* * * * *