USN □□□□□□□□□□

10CS63

Sixth Semester B.E. Degree Examination, Aug./Sept.2020

Compiler Design

Time: 3 hrs. Max. Marks:100

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

Important Note : 1. On completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages.
2. Any revealing of identification, appeal to evaluator and /or equations written eg, 42+8 = 50, will be treated as malpractice.

PART – A

1 a. Explain with a neat diagram the various phases of a compiler. Mention the input and output at each phase. **(08 Marks)**

b. What are the applications of compiler technology? Explain any one application in detail. **(07 Marks)**

c. Construct the transition diagram to recognize
(i) Relational operator in 'C' (ii) White space (iii) Keyword. **(05 Marks)**

2 a. With an example explain why left recursive grammar is not suitable for Top down parsing. **(05 Marks)**

b. Define FIRST and FOLLOW rules used in predictive parsing technique. **(06 Marks)**

c. Give the grammar

$S \rightarrow aB \mid aC \mid Sd \mid Se$

$B \rightarrow bBc \mid f$

$c \rightarrow g$

(i) Do the necessary changes to make it suitable for LL(1) parsing
(ii) Build the LL(1) parsing table. **(09 Marks)**

3 a. What is meant by handle pruning? Explain with the help of the grammar

$S \rightarrow SS+ \mid SS* \mid a$ and input string aaa* a++. **(06 Marks)**

b. Construct the LR(0) items for the following grammar

$S \rightarrow Ac$

$A \rightarrow AB \mid E$

$B \rightarrow aB \mid b$

Write an algorithm for constructing SLR parsing table and build the parsing table for the above grammar. **(14 Marks)**

4 a. Given the grammar

$S \rightarrow L = R \mid R$

$L \rightarrow *R \mid id$

$R \rightarrow L$

Find the canonical LR(1) items. **(10 Marks)**

b. Check weather the following grammar is LALR(1) or not. **(10 Marks)**

PART – B

5 a. Write a SDD for a simple desk calculator and show the annotated parse tree for

$1 * 2 * 3 * (4 + 5)n$ **(08 Marks)**

b. Give the L attributed definition for constructing the syntax tree during topdown parsing. Also show the dependency graph for a – 4 + c. **(08 Marks)**

c. Give the SDD for simple type declaration. **(04 Marks)**

6 a. Construct the DAG and identify the value number for the subexpression $a + b + (a + b)$ **(08 Marks)**

b. Translate the arithmetic expression $a = b * - c + b * - c$ into
(i) Syntax tree (ii) Quadruples (iii) Triples (iv) Indirect triples. **(08 Marks)**

c. Write a note on type checking. **(04 Marks)**

7 a. What is an activation record? Explain the purpose of each item in the activation record with example. Give the general structure of the activation record. **(07 Marks)**

b. What do you mean by calling sequence? Explain the actions performed during function call and return. **(07 Marks)**

c. Briefly explain time safety and performance metrics to be considered while designing a garbage collector. **(06 Marks)**

8 a. With an example explain common sub expression and deadcode elimination methods. **(10 Marks)**

b. What are the basic blocks and how do you partation a three address code into basic block. **(05 Marks)**

c. Explain the code generator algorithm. **(05 Marks)**

* * * * *