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

RV COLLEGE OF ENGINEERING®

(An Autonomous Institution affiliated to VTU)

V Semester B. E. Examinations March / April 2023

Information Science and Engineering

COMPILER DESIGN

Time: 03 Hours ***Maximum Marks: 100***

Instructions to candidates:

1. Answer all questions from Part A. Part A questions should be answered in first three pages of the answer book only.
2. Answer FIVE full questions from Part B. In Part B question number 2, 7 and 8 are compulsory. Answer any one full question from 3 and 4 & one full question from 5 and 6.

PART-A

1	1.1	Consider the production $S \rightarrow aBaC$, the number of "items" this production yields is _____.	01
	1.2	In $LR(1)$ item, the parameter 1 refers to the _____.	01
	1.3	While constructing SLR parsing table, if $A \rightarrow .a$ is in I_i and $GOTO(I_i, a) = I_j$, then the entry $ACTION[i, a]$ must be set as _____.	01
	1.4	A parse tree showing the values of attributes at each node is called _____.	01
	1.5	If $S1$ is the number of states of SLR parser for Grammar G and $S2$ is the number of states of $LALR$ parser for G. Then the relationship between $S1$ and $S2$ is _____.	01
	1.6	If one of the state of $LR(1)$ automata has 2 items: $A \rightarrow .x, c \mid d$ $B \rightarrow r., x \mid \$$ Then CLR parser has _____ conflict.	01
	1.7	In a compiler, keywords of the language are recognized during _____ phase.	01
	1.8	A top-down parser uses _____ derivation while parsing an input string.	01
	1.9	The _____ allocates and de-allocates space within heap.	01
	1.10	_____ are local code-improving transformations that can be applied to a program, usually through a sliding window.	01
	1.11	Given the following expression grammar: $E \rightarrow E * F \mid F + E \mid F$ $F \rightarrow F - F \mid id$ Compare the precedence of + and – operators.	02
	1.12	What are viable prefixes?	02
	1.13	Find the $FIRST(X)$ and $FOLLOW(X)$ in the following grammar. $S \rightarrow XYS, \quad S \rightarrow d, \quad Y \rightarrow \varepsilon, \quad X \rightarrow Y, \quad Y \rightarrow c, \quad X \rightarrow a$	02
	1.14	Consider the two regular expressions. $r = a^* + b^*$ $s = ab^* + ba^* + b^*a + (a^*b)^*$ i) Find a string corresponding to r but not to s. ii) Find a string corresponding to s but not to r.	02
	1.15	What is panic-mode error recovery?	02

PART-B

2	a	Explain the various phases of compiler and show the output for each phase for the expression $a = b + c * 25$.	10
	b	Explain the input buffering strategy used in lexical analysis phase.	06
3	a	Give the general formula to remove left recursion. Using the same, eliminate left recursion from the following grammar. A→ $Ba\|Aa\|c$ $B \rightarrow Bb\|Ab\|d$	04
	b	Explain with example the conflicts that may occur during shift reduce parsing.	04
	c	Given the grammar. $E \rightarrow TX$ $T \rightarrow (E)\| int\ Y$ $X \rightarrow +E\| \varepsilon$ $Y \rightarrow * T\| \varepsilon$ i) Construct first ad follow sets. ii) Construct $LL(1)$ parsing table.	08
		OR	
4	a	With a neat diagram, explain the role of parser.	04
	b	Consider the following augmented grammar. $S \rightarrow CC$ $C \rightarrow cC\|d$ i) Construct DFA of $LR(1)$ items for this grammar. ii) Construct the general $LALR(1)$ parsing table.	08
	c	Show that the following grammar is ambiguous. Write an equivalent unambiguous grammar. $E \rightarrow E + E \mid E * E \mid (E) \mid id$	04
5	a	Consider the grammar. $E \rightarrow E + T\|T$ $T \rightarrow T * F\| F$ $F \rightarrow (E)\| digit$ Write the SDD, annotated parse tree and dependency graph for the word $3 * 5 + 4 * 6$.	05
	b	Write the SDD to generate the three address code for an assignment statement S which is defined using the following grammar and use the SDD to translate the assignment statement $a = b + -c$ into three address code sequences. $S \rightarrow id = E;$ $E \rightarrow E + E\|-E\|(E)\|id$	06
	c	Construct the DAG for the following expressions: i) $a * (b - c) + (b - c) + d$ ii) $a + a + (a + a + a + (a + a + a + a))$	05
		OR	
6	a	Give the SDD to process a sample variable declaration D in 'C' language consisting of a basic type T followed by a list L of identifiers. T can be int or float. Construct the dependency graph for the input. $float\ a, b, c$	08

	b	Explain the following with an example: i) Three address codes ii) Quadruples.	08
7	a	What is the drawback of access-link approach to nonlocal data access? Explain with example how "display" provides a more efficient implementation of non-local data access.	08
	b	Explain the issues in code generation.	08
8	a	What are basic blocks and flow graphs. Write an algorithm to partition three address instructions into basic blocks.	06
	b	Explain with example, the following code optimization methods: i) Finding local common sub-expression ii) Dead code elimination.	06
	c	Simplify the following three address code: $d = b + c$ $e = a + b$ $b = b * c$ $a = e - d$ Assuming i) Only a is live on exit from the block ii) a, b and c are live on exit from the block.	04