Seat No.: _____ Enrolment No._____

GUJARAT TECHNOLOGICAL UNIVERSITY

BE - SEMESTER–VII (OLD) EXAMINATION – SUMMER 2019

Subject Code: 170701 **Date: 18/05/2019**

Subject Name: Compiler Design

Time: 02:30 PM TO 05:00 PM **Total Marks: 70**

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q.1	**(a)**	Explain different phases of compiler.	**07**
	(b)	(i) What is a symbol table? Discuss the most suitable data structure for it by stating merits / demerits.	**04**
		(ii) Explain linker & loader.	**03**
Q.2	**(a)**	What is the difference between parse tree and syntax tree? Write appropriate grammar and draw parse as well as syntax tree for a*(a-a^a).	**07**
	(b)	Construct a DFA without constructing NFA for following regular expression. Find minimized DFA. : (a | b)*a	**07**
		OR	
	(b)	Construct NFA for following Regular Expression : (a | b)*abb.	**07**
Q.3	**(a)**	(i) Compare top-down and bottom-up parser.	**07**
		(ii) Explain right-most-derivation-in-reverse with the help of an example.	
	(b)	What is left recursion? Eliminate the left recursion from the following grammar.	**07**
		$E \rightarrow E + T \mid T$ $T \rightarrow T * F \mid F$ $F \rightarrow (E) \mid id$	
		OR	
Q.3	**(a)**	Construct predictive parsing table for following.	**07**
		$S \rightarrow A$ $A \rightarrow aB \mid Ad$ $B \rightarrow bBC \mid f$ $C \rightarrow g$	
	(b)	Explain Operator precedence Parsing technique in detail.	**07**
Q.4	**(a)**	What is an activation record? Explain how they are used to access various local and global variables.	**07**
	(b)	Explain: Error Recovery Strategies in Compiler in brief	**07**
		OR	
Q.4	**(a)**	Translate the expression –(a+b)*(c+d)+(a+b+c) into 1. Quadruples 2. Triples 3. Indirect triples	**07**
	(b)	Explain various code optimization technique	**07**
Q.5	**(a)**	Write a brief note on input buffering techniques	**07**
	(b)	Define: DAG. Explain DAG representation of basic block with example.	**07**
		OR	
Q.5	**(a)**	Discuss issues in the design of code generation	**07**
	(b)	Explain Peephole Optimization in detail.	**07**

****