Seat No.: _____   Enrolment No._____

GUJARAT TECHNOLOGICAL UNIVERSITY
BE – SEMESTER- VII EXAMINATION-SUMMER 2023
Subject Code: 3170701   **Date: 22/06/2023**
Subject Name: Compiler Design
Time: 10:30 AM TO 01:00 PM   **Total Marks: 70**
Instructions:
1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**
4. **Simple and non-programmable scientific calculators are al.**

			MARKS
Q.1	**(a)**	Define lexemes, patterns and tokens	**03**
	(b)	Write a regular definition for:	**04**

      1. The language of all strings containing at least one 0 and at least one 1.
      2. The language of all strings that do not end with 01.
      3. The language of all strings containing 0's and 1's both are even.
      4. The language of all strings which are starting with 1 and ending with 0.

	(c)	Explain phases of compilers with suitable example.	**07**
Q.2	**(a)**	Differentiate ambiguous and unambiguous grammar.	**03**
	(b)	Write a short note on input buffering techniques.	**04**
	(c)	Construct a DFA for a given regular expression using syntax tree with firstpos, lastpos and followpos function.	**07**

<div align="center">(a|b)*a</div>

<div align="center">**OR**</div>

	(c)	Construct the NFA for following regular expression using Thompson's construction. Apply subset construction method to convert into DFA.	**07**

<div align="center">(a+b)*abb#</div>

Q.3	**(a)**	Give the translation scheme that convert infix to postfix notation.	**03**
	(b)	Define: Left Recursive Grammar. Check Following grammar is left recursive or not, if yes, remove left recursive from it.	**04**

A → ABd / Aa / a

B → Be / b

	(c)	Construct an SLR Parsing table for the following grammar.	**07**

S → Aa | bAc | bBa
A → d
B → d

<div align="center">**OR**</div>

Q.3	**(a)**	Explain shift reduce parsing technique in brief.	**03**
	(b)	Differentiate top down parsing and bottom up parsing.	**04**
	(c)	Show that the following grammar	**07**

S → AaAb | BbBa
A → ε
B → ε
 is LL(1) but not SLR(1).

Q.4	**(a)**	Write a short note on activation record.	**03**
	(b)	Explain Symbol table management in brief.	**04**
	(c)	List and explain various storage allocation strategies.	**07**

OR

Q.4	**(a)**	What do you mean by dangling references?	**03**
	(b)	Explain Error Recovery Strategies in Compiler in brief.	**04**
	(c)	Write three address code for	**07**

$$a = b*-c + b*-c$$

| Q.5 | **(a)** | Translate the arithmetic expression $(a + b) * (c + d)$ into | **03** |

    1. Syntax tree

    2. Postfix notation

| | **(b)** | Compare: Static v/s Dynamic Memory Allocation | **04** |
| | **(c)** | Explain various code optimization techniques. | **07** |

OR

| Q.5 | **(a)** | Construct a DAG for the following expression: | **03** |

$$a = b*-c + b*-c$$

| | **(b)** | Explain various parameter passing methods. | **04** |
| | **(c)** | Explain various issues in design of code generator. | **07** |

****