

GUJARAT TECHNOLOGICAL UNIVERSITY**BE- SEMESTER-VII (NEW) EXAMINATION – WINTER 2020****Subject Code:2170701****Date:19/01/2021****Subject Name:Compiler Design****Time:10:30 AM TO 12:30 PM****Total Marks: 56****Instructions:**

1. Attempt any FOUR questions out of EIGHT questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

		MARKS
Q.1	(a) Compare and contrast compilers and interpreters.	03
	(b) Write a short note on LEX Tool.	04
	(c) Explain phases of compilers with suitable example.	07
Q.2	(a) Define terms: pattern, lexeme, token	03
	(b) Write any one method used in lexical analyzer for buffering the input.	04
	(c) Construct nondeterministic finite automata by using Thomson's Construction for following regular expression. Show the sequence of moves made in processing the input string ababbab. (a b)* abb	07
Q.3	(a) Consider the following grammar to construct leftmost and right most derivation for the sentence abab.	03
	$S \rightarrow aSbS bSaS \epsilon$	
	(b) Write rule(s) to check grammar is left recursive or not. Remove left recursive from the following grammar. $\begin{aligned} S &\rightarrow aBDh \\ B &\rightarrow Bb c \\ D &\rightarrow EF \\ E &\rightarrow g \epsilon \\ F &\rightarrow f \epsilon \end{aligned}$	04
	(c) Construct the SLR parsing table for the following Grammar. $\begin{aligned} E &\rightarrow E+T T \\ T &\rightarrow T*F F \\ F &\rightarrow (E) id \end{aligned}$	07
Q.4	(a) Write rule(s) to check grammar is operator grammar or not.	03
	(b) Explain any two error-recovery strategies.	04
	(c) Construct LALR parsing table for the following grammar.	07

$S \rightarrow CC$
 $C \rightarrow cC$
 $C \rightarrow d$

- Q.5** (a) Define : 1) synthesized attribute 2) inherited attribute **03**
 (b) Explain any two parameter passing methods. **04**
 (c) Write SDD for arithmetic expression and Construct annotated parse tree for the input expression $(4*7+1)*2$. **07**
- Q.6** (a) What is activation record? **03**
 (b) What is Intermediate form of the code? What are the advantages of it? **04**
 (c) Construct syntax directed translation scheme and definition that translate arithmetic operation from infix to postfix in which an operator appears before its operand as for example xy- is a positive notation for x-y . Give annotated parse tree for the input $9-5+2$ and $9-5*2$. **07**
- Q.7** (a) Explain any two issues in design of code generator **03**
 (b) Compare: static v/s dynamic Memory Allocation. **04**
 (c) Translate the expression $-(a+b)*(c+d)+(a+b+c)$ into **07**
 1. Quadruples 2. Triples 3. Indirect triples
- Q.8** (a) What are the limitations of static storage allocation? **03**
 (b) Write a short note on Symbol table management. **04**
 (c) Write any two methods of code-optimization in detail. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER- VIII EXAMINATION – SUMMER 2020****Subject Code: 2170701****Date: 26/10/2020****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.

		MARKS
Q.1	(a) Write a brief note on input buffering techniques.	03
	(b) Explain with suitable example what is bootstrapping?	04
	(c) Explain different phases of compiler.	07
Q.2	(a) Explain panic mode recovery strategy.	03
	(b) Check whether the following grammar is ambiguous or not. $S \rightarrow (S) S$ $S \rightarrow \epsilon$	04
	(c) Construct a DFA for a given regular expression $(010+00)^*(10)^*$	07
	OR	
	(c) Construct DFA accepting the strings of binary digits which are even numbers.	07
Q.3	(a) Define the following terms: 1) Handle 2) Handle pruning 3) Left Factoring	03
	(b) Explain shift reduce parsing technique in brief.	04
	(c) Construct an LL(1) parsing table for the following grammar: $S \rightarrow aBDh$ $B \rightarrow cC$ $C \rightarrow bC \epsilon$ $D \rightarrow EF$ $E \rightarrow g \epsilon$ $F \rightarrow f \epsilon$	07
	OR	
Q.3	(a) What is a symbol table? Discuss any two data structures suitable for it.	03
	(b) Eliminate left recursion from following grammar. $S \rightarrow Aa b$ $A \rightarrow Ac Sd f$	04
	(c) Construct an SLR Parsing table for the following grammar. $E \rightarrow E + T T$ $T \rightarrow T * F F$ $F \rightarrow (E)$ $F \rightarrow id$	07

P.T.O

- Q.4** (a) Explain Control Stack. **03**
 (b) What do you mean by dangling references? **04**
 (c) Translate the expression $-(a + b) * (c + d) + (a + b + c)$ into **07**
 1. Quadruples
 2. Triples
 3. Indirect triples.

OR

- Q.4** (a) Translate the arithmetic expression $a * - (b + c)$ into **03**
 1. Syntax Tree
 2. Postfix Notation
 3. Three Address code
 (b) Write three address code for **04**
 $a := a + a * b + a * b * c - a/b + a * b$
 (c) What is the use of a symbol table? How the identifiers are stored in the symbol table? **07**
- Q.5** (a) Explain parameter passing techniques for procedure. **03**
 (b) List the different conflicts that occur in Bottom up parsing and give examples for that. **04**
 (c) Explain various code optimization techniques. **07**

OR

- Q.5** (a) Draw a DAG for expression: $a + a * (b - c) + (b - c) * d$. **03**
 (b) Compare inherited attributes vs. synthesized attributes. **04**
 (c) Write the generic issues in the design of code generators. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER– VII (New) EXAMINATION – WINTER 2019****Subject Code: 2170701****Date: 23/11/2019****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.

		MARKS
Q.1	(a) Explain tokens, lexemes, Pattern with example.	03
	(b) Distinguish between ambiguous and unambiguous grammar?	04
	(c) Explain the analysis synthesis model of compilation. List the factors that affect the design of compiler. Also List major functions done by compiler.	07
Q.2	(a) Write a regular definition for	03
	1. The Language of all strings that do not end with 01.	
	2. All strings of digit that contain no leading 0's.	
	(b) Explain backtracking with example.	04
	(c) Construct a DFA for a given regular expression (a b)*abb.	07
	OR	
	(c) Construct DFA without constructing NFA for following regular expression: $a^*b^*a(a b)b^*a\#$.	07
Q.3	(a) Perform the Left factoring of following Grammar.	03
	$S \rightarrow iEtS / iEtSeS / a \quad E \rightarrow b$	
	(b) Write a brief note on input buffering techniques.	04
	(c) Explain Recursive Descent Parser with example.	07
	OR	
Q.3	(a) Explain the following:	03
	1. Handle	
	2. Forward Reference	
	3. Conflicts in LR Parsing	
	(b) Explain non-recursive predictive parsers. Draw the block diagram of it.	04
	(c) Generate the SLR parsing table for the following Grammar.	07
	$S \rightarrow Aa bAc bBa$	
	$A \rightarrow d$	
	$B \rightarrow d$	
Q.4	(a) Define attributed grammar? Which phase of the compilation process does it facilitate? Explain with example.	03
	(b) Explain Stack Allocation and Activation Record Organization in brief.	04
	(c) Write down steps to set precedence relationship for Operator Precedence Grammar. Design precedence table for: $E \rightarrow E+E E * E E^E id$.	07
	OR	
Q.4	(a) Construct a DAG for $(a+b)^* (a+b+c)$.	03
	(b) Explain Error Recovery Strategies in Compiler in brief.	04
	(c) Show syntax directed definition for simple desk calculator. Also show annotated parse tree for $3*5+4n$, where n indicates newline.	07
Q.5	(a) Differentiate: static v/s dynamic memory allocations.	03
	(b) Discuss symbol table management in detail.	04

- (c) Translate following arithmetic expression $(a * b) + (c + d) - (a + b)$ into **07**
1] Quadruples
2] Triple
3] Indirect Triple

OR

- Q.5** (a) Explain any three code optimization techniques with example. **03**
(b) Explain various parameter passing methods. **04**
(c) Explain various issues in design of code generator. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII(NEW) EXAMINATION – SUMMER 2019****Subject Code: 2170701****Date: 10/05/2019****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.

		MARKS
Q.1	(a) Define lexemes, patterns and tokens.	03
	(b) Differentiate compilers and interpreters.	04
	(c) Explain analysis of source program for compilers.	07
Q.2	(a) Give regular definition for signed and unsigned numbers.	03
	(b) Check whether the following grammar is ambiguous or not. $S \rightarrow (S) S$ $S \rightarrow \epsilon$	04
	(c) Draw DFA from regular expression without constructing NFA. $(a b c)^* a (b c)^* \#$	07
	OR	
	(c) Draw NFA from regular expression using Thomson's construction and convert it into DFA. $(a b)^* a b^* a$	07
Q.3	(a) Define handle and handle pruning.	03
	(b) Construct operator precedence relations table for following grammar. $E \rightarrow E + E$ $E \rightarrow E - E$ $E \rightarrow E * E$ $E \rightarrow (E)$ $E \rightarrow id$ Assume suitable operator associativity and precedence.	04
	(c) Construct recursive descent parser for following grammar. $E \rightarrow T A$ $A \rightarrow + T A$ $A \rightarrow \epsilon$ $T \rightarrow F B$ $B \rightarrow * F B$ $B \rightarrow \epsilon$ $F \rightarrow (E)$ $F \rightarrow id$	07
	OR	
Q.3	(a) Differentiate top down parsing and bottom up parsing.	03
	(b) Construct syntax directed translation scheme for infix to postfix conversion.	04
	(c) Construct LL(1) parsing table for following grammar. Check whether the grammar is LL(1) or not. $A \rightarrow A a B$ $A \rightarrow x$ $B \rightarrow B C b$ $B \rightarrow C y$ $C \rightarrow C c$ $C \rightarrow \epsilon$	07

- Q.4** (a) Define **Intermediate code and its importance**. **03**
 (b) Construct **LR(0) item** sets for following grammar. **04**
 $S \rightarrow AaAb$
 $S \rightarrow BbBa$
 $A \rightarrow \epsilon$
 $B \rightarrow \epsilon$
 (c) Explain various **error recovery schemes in detail**. **07**
- OR**
- Q.4** (a) **Differentiate LR(1) and LALR(1) parsers**. **03**
 (b) Construct syntax tree and **DAG** for following expression. **04**
 $a = (b+c+d) * (b+c-d) + a$
 (c) Explain **quadruples, triples and indirect triples** with examples. **07**
- Q.5** (a) Define basic block with a simple example. **03**
 (b) Explain activation record. **04**
 (c) Explain various methods of peephole optimization. **07**
- OR**
- Q.5** (a) Explain static storage allocation. **03**
 (b) Explain any **two parameter passing methods**. **04**
 (c) Explain various **issues in design of code generator**. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII (NEW) EXAMINATION – WINTER 2018****Subject Code: 2170701****Date: 15/11/2018****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.

		MARKS
Q.1	(a) List the cousins of compiler and explain the role of any one of them.	03
	(b) Write a brief note on input buffering techniques.	04
	(c) Explain front end and back end of compiler in detail.	07
Q.2	(a) Define the following terms and give suitable example for it. 1) Handle 2) Handle pruning 3) Left Factoring	03
	(b) Explain all error recovery strategies used by parser.	04
	(c) Construct LL(1) parsing table for the following Grammar: E -> E+T T T -> T*F F F -> (E) a	07
	OR	
	(c) Construct NFA for following regular expression using Thompson's notation and then convert it into DFA. (a/b)*abb#	07
Q.3	(a) Define the following terms and give suitable example for it. 1) Augmented Grammar 2) LR(0) Item 3) LR(1) Item	03
	(b) Differentiate Top Down Parsing and Bottom up parsing	04
	(c) Construct SLR parsing table for the following grammar : S -> (L) a L -> L,S S	07
	OR	
Q.3	(a) Give the difference between SLR and CLR Parser.	03
	(b) List the different conflicts that occur in Bottom up parsing and give examples for that.	04
	(c) Implement the following grammar using Recursive Descent Parser. S -> Aa bAc bBa A -> d B -> d	07
Q.4	(a) What is Ambiguous Grammar? Describe with example.	03
	(b) Give the difference between synthesized attributes and inherited attributes	04
	(c) Construct CLR parsing table for the following grammar : S -> AA A -> aA b	07
	OR	
Q.4	(a) List the different issues in code generation phase and describe any two issues.	03
	(b) Explain parameter passing techniques for procedure.	04
	(c) Explain Quadruple, triple and indirect triple with suitable example.	07

- Q.5** (a) Draw syntax tree and DAG for the statement $x=(a+b)*(a+b+c)*(a+b+c+d)$ **03**
(b) Explain dynamic memory allocation strategy. **04**
(c) What is an activation record? Explain how they are used to access local and global variables. **07**

OR

- Q.5** (a) Write a note on stack allocation strategy. **03**
(b) Give the translation scheme that converts infix to postfix expression for the following grammar and also generate the annotated parse tree for input string "id+id*id"
 $E \rightarrow E+T \mid T$ **04**
 $T \rightarrow T * F \mid F$
 $F \rightarrow id$
(c) Discuss various code optimization techniques with examples. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY
BE IV - SEMESTER VII (NEW SYLLABUS) EXAMINATION- SUMMER - 2018

Subject Code: 2170701**Date: 28/04/2018****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) Define cross compiler, token and handle. 03
 (b) Draw transition diagram for relational operators. 04
 (c) Explain phases of compiler with example. 07

- Q.2** (a) Explain panic mode recovery strategy. 03
 (b) Write a short note on input buffering method. 04
 (c) Explain subset construction method with example. 07

OR

- (c) Draw DFA for the following regular expression using firstpos(), lastpos() and followpos() functions. 07
 $(a | b)^* a$
- Q.3** (a) What is operator grammar? Check the following grammar is operator or not. Justify your answer. 03
 $E \rightarrow EOE$
 $E \rightarrow id$
 $O \rightarrow * | + | -$
- (b) Check the following grammar is left recursive or not. Justify your answer. If Left recursive then make grammar as non-left recursive. 04
 $S \rightarrow (L) | a$
 $L \rightarrow L, S | S$
- (c) Construct CLR parsing table for the following grammar. 07
 $S \rightarrow CC$
 $C \rightarrow cC | d$

OR

- Q.3** (a) Consider the following grammar and construct the corresponding left most and right most derivations for the sentence abab. 03
 $S \rightarrow aSbS | bSaS | \epsilon$
- (b) Find out FIRST and FOLLOW for the following grammar. 04
 $S \rightarrow 1AB | \epsilon$
 $A \rightarrow 1AC | 0C$
 $B \rightarrow 0S$
 $C \rightarrow 1$
- (c) Explain SLR parsing method with example. 07
- Q.4** (a) What is symbol table? For what purpose, compiler uses symbol table? 03
 (b) Write a short note on activation record. 04
 (c) Write syntax directed definition for simple desk calculator. Using this definition, draw annotated parse tree for $3*5+4n$. 07

OR

Q.4	(a) Explain algebraic simplifications and flow of control optimization characteristics of peephole optimization.	03
	(b) Explain Quadruples and Triples form of three address code with example.	04
	(c) What is inherited attribute? Write syntax directed definition with inherited attributes for type declaration for list of identifiers.	07
Q.5	(a) Draw a DAG for expression: $a + a * (b - c) + (b - c) * d$.	03
	(b) Compare: Static v/s Dynamic Memory Allocation.	04
	(c) Explain any three code optimization methods.	07
OR		
Q.5	(a) Write difference(s) between stack and heap memory allocation.	03
	(b) Explain any two methods of parameter passing.	04
	(c) Explain various issues in design of code generator.	07

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII (NEW) EXAMINATION – WINTER 2017****Subject Code: 2170701****Date: 02/11/2017****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.

		MARKS
Q.1	(a) What is the difference between compiler and interpreter?	03
	(b) Explain analysis phase of the source program with example.	04
	(c) Write an algorithm for Thompson's construction method. Apply the algorithm to construct NFA for following regular expression. $(a \mid b)^*abb$.	07
Q.2	(a) What is a pass in a compiler? What is the effect of reducing the number of passes?	03
	(b) Explain error recovery strategies used by parser.	04
	(c) What is operator grammar? Generate precedence function table for following grammar. $E \rightarrow EAE \mid id$ $A \rightarrow + \mid *$	07
	OR	
	(c) Define handle and handle pruning. Explain the stack implementation of shift reduce parser with the help of example.	07
Q.3	(a) Give the translation scheme that converts infix to postfix notation. Generate the annotated parse tree for input string 3-5+4.	03
	(b) Explain buffer pairs and sentinels.	04
	(c) Check given grammar is LL(1) but not SLR(1). $S \rightarrow AaAb \mid BbBa$ $A \rightarrow \epsilon$ $B \rightarrow \epsilon$	07
	OR	
Q.3	(a) Write a rule of Left factoring a grammar and give example.	03
	(b) Explain role of lexical analyzer.	04
	(c) Define syntax tree. What is s-attributed definition? Explain construction of syntax tree for the expression $a-4+c$ using SDD.	07
Q.4	(a) Translate the arithmetic expression $a^*-(b+c)$ into 1. Syntax tree 2. Postfix notation 3. Three address code	03
	(b) Write Syntax Directed Definition to produce three address code for the expression containing the operators $:=, +, -, (\text{unary minus}), ()$ and id .	04
	(c) What is activation record? Explain stack allocation of activation records using example.	07
	OR	
Q.4	(a) What is activation tree?	03
	(b) Explain parameter passing techniques for procedure.	04
	(c) What is importance of intermediate code? Discuss various representations of three address code using the given expression. $a = b * -c + b * -c$.	07

- Q.5** (a) Explain three loop optimization techniques with example. **03**
(b) What is code optimization? Explain data flow equation. **04**
(c) Describe code generator design issues. **07**
- OR**
- Q.5** (a) Define following : DAG, Basic Blocks, Flow graph **03**
(b) Explain peephole optimization. **04**
(c) Explain function preserving transformations with example. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII (NEW) - EXAMINATION – SUMMER 2017****Subject Code: 2170701****Date: 29/04/2017****Subject Name: Compiler Design****Time: 02.30 PM to 05.00 PM****Total Marks: 70****Instructions:**

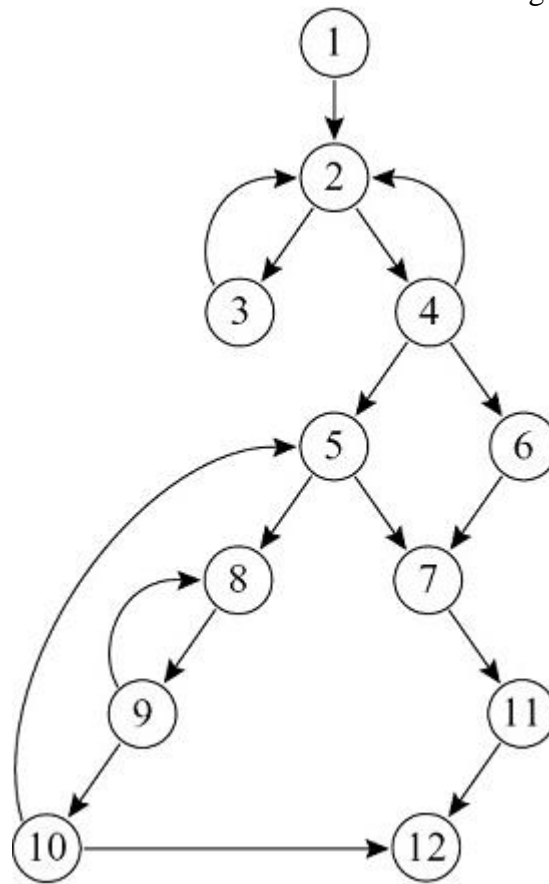
1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

- Q.1** (a) Explain Semantic analysis and Syntax analysis phases of compiler with suitable example. Also explain the errors generated by these two phases. **07**
- (b) Construct the NFA using thompson's notation for following regular expression and then convert it to DFA. **07**
 $(a / b)^* ab\#$
- Q.2** (a) Check following grammar is LL (1) or not? **07**
 $S \rightarrow aB \mid \epsilon$
 $B \rightarrow bC \mid \epsilon$
 $C \rightarrow cS \mid \epsilon$
- (b) What is left factoring and left recursion? Explain it with suitable example. **07**
- OR**
- (b) Construct CLR parsing table for following grammar. **07**
 $S \rightarrow aSA \mid \epsilon$
 $A \rightarrow bS \mid c$
- Q.3** (a) Show that following grammar is not a SLR (1) grammar. **07**
 $S \rightarrow AaBa \mid BbBa$
 $A \rightarrow \epsilon$
 $B \rightarrow \epsilon$
- (b) Develop a syntax directed definition for following grammar. **07**
 $E \rightarrow TE'$
 $E' \rightarrow +TE' \mid \epsilon$
 $T \rightarrow (E)$
 $T \rightarrow id$
- OR**
- Q.3** (a) Write a grammar to declare variables with data type int or float or char. Also develop a syntax directed definition for that. Draw the dependency graph for same. **07**
- (b) Define operator precedence grammar. Construct precedence matrix and precedence graph for arithmetic grammar as shown below: **07**
 $E \rightarrow E + T \mid T$
 $T \rightarrow T * F \mid F$
 $F \rightarrow (E) \mid id$
- Q.4** (a) Explain Activation record and Activation tree in brief. **07**
- (b) Explain Quadruple, triple, and indirect triple with suitable example. **07**
- OR**
- Q.4** (a) Write a note on peephole optimization. **07**
- (b) Write a short note on symbol table management. **07**
- Q.5** (a) Define a following: Basic block, Constant folding, Natural loop, Handle **07**

- (b) Construct DAG for $a + a * (b - c) + (b - c) * d$. also generate three address code for same. 07

OR

- Q.5 (a) Discuss the issues in the design of code generation. 07
(b) Define dominators. Construct dominator tree for following graph. 07



GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII(NEW) • EXAMINATION – WINTER 2016****Subject Code:2170701****Date:18/11/2016****Subject Name:Compiler Design****Time:10.30 AM to 1.00 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) Define the following terms: 07

1. Token
2. Pattern
3. Lexeme
4. Ambiguous grammar
5. Handle pruning
6. Compiler
7. DAG

(b) Discuss various error recovery strategies of compiler. 07**Q.2 (a) Write a short note on input buffering methods. 07****(b) Explain subset construction method for constructing DFA from an NFA with an example. 07****OR****(b) Construct DFA for the following regular expression using syntax tree with firstpos, lastpos and followpos function. 07** $(a | b)^*a$ **Q.3 (a) Explain SLR parsing method with example. 07****(b) Construct LL(1) Parsing table for the following grammar. Also show moves made by input string : abba. 07** $S \rightarrow aBa$ $B \rightarrow bB \mid \epsilon$ **OR****Q.3 (a) Check that following grammar is LALR or not. 07** $S \rightarrow L=R$ $S \rightarrow R$ $L \rightarrow *R$ $L \rightarrow id$ $R \rightarrow L$ **(b) Write a short note on operator precedence parsing with an example. 07****Q.4 (a) Write S-attributed syntax directed definition for simple desk calculator. Draw annotated parse tree for any valid input. 07****(b) What is Intermediate Code? What is its importance? Discuss various representations of three address code. 07****OR****Q.4 (a) Discuss synthesized attributes and inherited attributes in details. 07****(b) Explain Peephole Optimization method. 07****Q.5 (a) Discuss generic issues in the design of code generator. 07**

(b) Explain the following parameter passing methods.

07

1. Call-by-value
2. Call-by-reference
3. Copy-Restore
4. Call-by-Name

OR

Q.5 (a) Explain Dynamic storage allocation technique.

07

(b) Discuss any three methods for code optimization.

07

Seat No.: _____

Enrolment No. _____

GUJARAT TECHNOLOGICAL UNIVERSITY

BE - SEMESTER-VII (OLD) - EXAMINATION – SUMMER 2018

Subject Code:170701

Date:08/05/2018

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)** Define following terms: **07**
- i. Compiler
 - ii. Interpreter
 - iii. Assembler
 - iv. Regular Expression
 - v. Token
 - vi. Lexeme
 - vii. Pattern
- (b)** How do the parser and scanner communicate? Explain with the block diagram communication between them. Also explain: What is input buffering? **07**
- Q.2 (a)** Construct DFA for following regular expression without constructing NFA and optimize the same. **07**
- $(a | \epsilon)^* a b (a | b)^* \#$
- (b)** Discuss various Storage allocation strategies in detail. **07**
- OR**
- (b)** Explain various data structures used in symbol table management. **07**
- Q.3 (a)** Construct a SLR parsing table for following grammar. **07**
- $S \rightarrow aAb | bB$
 $A \rightarrow Aa | \epsilon$
 $B \rightarrow Bb | \epsilon$
- (b)** Check whether the given grammar is LL (1) or not? **07**
- $S \rightarrow aAC | bB$
 $A \rightarrow Abc | Abd | e$
 $B \rightarrow f | g$
 $C \rightarrow h | i$
- OR**
- Q.3 (a)** Construct the LALR parsing table for the following grammar. **07**
- $S \rightarrow CC$
 $C \rightarrow aC$
 $C \rightarrow d$
- (b)** Write a syntax directed definition for desk calculator. Justify whether this is an S-attributed definition or L-attributed definition. Using this definition draw annotated parse tree for $3*5+4n$. **07**
- Q.4 (a)** What is intermediate code? Explain different types of intermediate code representations. Also discuss importance of intermediate code. **07**
- (b)** Explain activation tree, control stack, the Scope of Declaration and Bindings of Names. **07**
- OR**
- Q.4 (a)** Explain Operator precedence Parsing technique in detail. **07**
- (b)** Write down short note on Error – Recovery Strategies. **07**
- Q.5 (a)** i. Construct the DAG for the following basic block: **07**

d: = b * c

e: = a + b

b: = b * c

a: = e - d

ii. Describe issues in code generation process.

(b) Explain Peephole Optimization in detail.

07

OR

Q.5 (a) Write Short notes on

07

i. Local and loop optimization

ii. induction variable elimination

(b) Explain automatic generation of lexical analyzer and parser.

07

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII (NEW) - EXAMINATION – SUMMER 2017****Subject Code: 2170701****Date: 29/04/2017****Subject Name: Compiler Design****Time: 02.30 PM to 05.00 PM****Total Marks: 70****Instructions:**

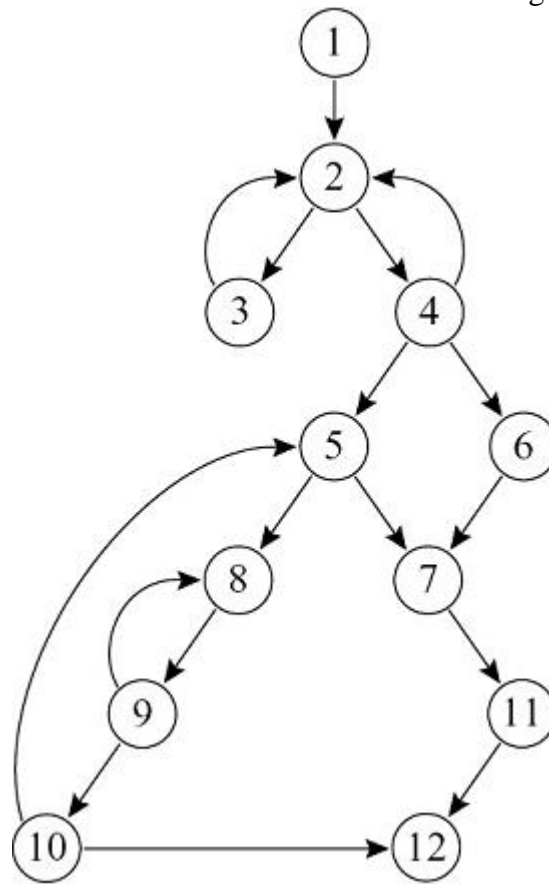
1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

- Q.1** (a) Explain Semantic analysis and Syntax analysis phases of compiler with suitable example. Also explain the errors generated by these two phases. **07**
 (b) Construct the NFA using thompson's notation for following regular expression and then convert it to DFA. **07**
 $(a / b)^* ab\#$
- Q.2** (a) Check following grammar is LL (1) or not? **07**
 $S \rightarrow aB \mid \epsilon$
 $B \rightarrow bC \mid \epsilon$
 $C \rightarrow cS \mid \epsilon$
 (b) What is left factoring and left recursion? Explain it with suitable example. **07**
- OR**
- (b) Construct CLR parsing table for following grammar. **07**
 $S \rightarrow aSA \mid \epsilon$
 $A \rightarrow bS \mid c$
- Q.3** (a) Show that following grammar is not a SLR (1) grammar. **07**
 $S \rightarrow AaBa \mid BbBa$
 $A \rightarrow \epsilon$
 $B \rightarrow \epsilon$
 (b) Develop a syntax directed definition for following grammar. **07**
 $E \rightarrow TE'$
 $E' \rightarrow +TE' \mid \epsilon$
 $T \rightarrow (E)$
 $T \rightarrow id$
- OR**
- Q.3** (a) Write a grammar to declare variables with data type int or float or char. Also develop a syntax directed definition for that. Draw the dependency graph for same. **07**
 (b) Define operator precedence grammar. Construct precedence matrix and precedence graph for arithmetic grammar as shown below: **07**
 $E \rightarrow E + T \mid T$
 $T \rightarrow T * F \mid F$
 $F \rightarrow (E) \mid id$
- Q.4** (a) Explain Activation record and Activation tree in brief. **07**
 (b) Explain Quadruple, triple, and indirect triple with suitable example. **07**
- OR**
- Q.4** (a) Write a note on peephole optimization. **07**
 (b) Write a short note on symbol table management. **07**
- Q.5** (a) Define a following: Basic block, Constant folding, Natural loop, Handle **07**

- (b) Construct DAG for $a + a * (b - c) + (b - c) * d$. also generate three address code for same. 07

OR

- Q.5 (a) Discuss the issues in the design of code generation. 07
(b) Define dominators. Construct dominator tree for following graph. 07



GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VII(NEW) • EXAMINATION – WINTER 2016****Subject Code:2170701****Date:18/11/2016****Subject Name:Compiler Design****Time:10.30 AM to 1.00 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) Define the following terms: **07**
1. Token
 2. Pattern
 3. Lexeme
 4. Ambiguous grammar
 5. Handle pruning
 6. Compiler
 7. DAG
- (b) Discuss various error recovery strategies of compiler. **07**
- Q.2** (a) Write a short note on input buffering methods. **07**
- (b) Explain subset construction method for constructing DFA from an NFA with an example. **07**
- OR**
- (b) Construct DFA for the following regular expression using syntax tree with firpos, laspos and followpos function. **07**
- $(a \mid b)^*a$
- Q.3** (a) Explain SLR parsing method with example. **07**
- (b) Construct LL(1) Parsing table for the following grammar. Also show moves made by input string : abba. **07**
- $S \rightarrow aBa$
 $B \rightarrow bB \mid \epsilon$
- OR**
- Q.3** (a) Check that following grammar is LALR or not. **07**
- $S \rightarrow L=R$
 $S \rightarrow R$
 $L \rightarrow *R$
 $L \rightarrow id$
 $R \rightarrow L$
- (b) Write a short note on operator precedence parsing with an example. **07**
- Q.4** (a) Write S-attributed syntax directed definition for simple desk calculator. Draw annotated parse tree for any valid input. **07**
- (b) What is Intermediate Code? What is its importance? Discuss various representations of three address code. **07**
- OR**
- Q.4** (a) Discuss synthesized attributes and inherited attributes in details. **07**
- (b) Explain Peephole Optimization method. **07**
- Q.5** (a) Discuss generic issues in the design of code generator. **07**

(b) Explain the following parameter passing methods.

07

1. Call-by-value
2. Call-by-reference
3. Copy-Restore
4. Call-by-Name

OR

Q.5 (a) Explain Dynamic storage allocation technique.

07

(b) Discuss any three methods for code optimization.

07
