

III B. Tech I Semester Regular Examinations, Dec/Jan-2022-23
COMPILER DESIGN

[Common to CSE (AIML), CSE(AI), CSE(DS), CSE(AIDS), AIDS, AIML, CSD]
 Time: 3 hours Max. Marks: 70

Answer any **FIVE** Questions **ONE** Question from **Each unit**

All Questions Carry Equal Marks

UNIT-I

1. a) Explain about input buffering schemes in lexical analysis. [7M]
 b) Write a regular expression for identifiers and reserved words. [7M]
 Design the transition diagrams for them.
 (OR)
2. a) Explain various phases of compiler with the example: [10M]
 $i:=i*70+j+2$.
 b) Construct DFA equivalent to regular expression $r= b (a + b)^* a (ab)^*$. [4M]

UNIT-II

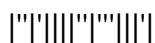
3. a) Construct LL(1) parsing table for the following grammar: [10M]
 $S \rightarrow L = R, S \rightarrow R, L \rightarrow *R, L \rightarrow id, R \rightarrow L$
 b) Define Ambiguous grammar? Explain it with an Example. [4M]
 (OR)
4. a) Construct the predictive parser for the following grammar [7M]
 $S \rightarrow (L) \mid a L \rightarrow L, S \mid S$.
 b) State and explain the rules used to compute first and follow functions with the help of $E \rightarrow E+T \mid T T \rightarrow T^*F \mid F F \rightarrow F^* \mid a \mid b$. [7M]

UNIT-III

5. a) What is an intermediate code? Explain different types of intermediate codes forms and represent the following statement in different forms: $W = (A+B) - (C+D) + (A + B + C)$ [7M]
 b) Differentiate between Synthesized and Inherited attributes with suitable examples. [7M]
 (OR)
6. a) Explain the structure of LR parses and various functions of it. [4M]
 b) $S \rightarrow L=R \mid R L \rightarrow * R \mid id R \rightarrow L$. For the given grammar construct SLR parse table. Explain Rules. [10M]

UNIT-IV

7. a) What is code optimization? Explain about various levels and types of optimizations. [7M]
 b) What is meant by activation of procedure? How it can be represented with activation tree and record? Explain with quick sort example. [7M]
 (OR)
8. a) What is a Flow Graph? Explain how a given program can be converted in to a Flow graph? [7M]
 b) With an example explain the following loop optimization techniques: (i) Code motion (ii) Induction variable elimination and (iii) strength reduction. [7M]



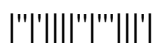
Code No: R2031421

R20

SET - 1

UNIT-V

9. a) Write the algorithm for a simple code generator. [7M]
b) Discuss about register allocation and assignment in target code generation. [7M]
- (OR)
10. a) Explain various issues that affect the efficiency of generated code. [7M]
b) What are the different object code forms in code generation and explain. [7M]



III B. Tech I Semester Regular Examinations, Dec/Jan-2022-23
COMPILER DESIGN

[Common to CSE(AIML), CSE(AI), CSE(DS), CSE(AIDS), AIDS, AIML,CSD]
 Time: 3 hours Max. Marks: 70

Answer any **FIVE** Questions **ONE** Question from **Each unit**
 All Questions Carry Equal Marks

UNIT-I

1. a) Explain how lex program will perform the lexical analysis for the following patterns in C: Identifiers, comments, numerical constants and arithmetic operators. [7M]
 - b) Construct NFA equivalent to regular expression $r = (a + b)^* ab$ ($aa+bb$) and convert it into DFA. [7M]
- (OR)
2. a) Consider the following Conditional statement: if ($x > 3$) then $y = 5$ else $y = 10$; How does lexical analyzer help the above statement in process of compilation? [7M]
 - b) Define Regular Expression. Explain about the Properties of Regular Expressions. Give examples. [7M]

UNIT-II

3. a) Check whether the grammar is LL (1) or not, and construct a predictive parsing table for following grammar: $S \rightarrow iEtSS_1 / a$, $S_1 \rightarrow eS / \epsilon$, $E \rightarrow b$. [7M]
 - b) Define Ambiguous Grammar? Check whether the grammar $S \rightarrow aAB$, $A \rightarrow bC/cd$, $C \rightarrow cd$, $B \rightarrow c/d$, is Ambiguous or not? [7M]
- (OR)
4. a) Parse the input string $intid,id$; using predictive parser for the grammar:
 $S \rightarrow TL$;
 $T \rightarrow int \mid float$
 $L \rightarrow L,id \mid id$ [7M]
 - b) What is Dangling ELSE ambiguity? How it can be solved with LR parsers? Explain with an example. [7M]

UNIT-III

5. a) Write the quadruple, triple, indirect triple for the expression $-(a*b) + (c+d) - (a+b+c+d)$. [7M]
 - b) For the grammar below: $E \rightarrow E + T \mid T$, $T \rightarrow num .num \mid num$ Give an SDD to determine the type of each term T and expression E. [7M]
- (OR)
6. a) Differentiate LR(0) and LR(1) items with examples [4M]
 - b) $S \rightarrow L=R \mid R L \rightarrow *R \mid id R \rightarrow L$ construct CLR(1) parser and explain the procedure. [10M]



UNIT-IV

7. a) What is a leader of basic block? Write and explain the algorithm used to find leaders. Draw flow graph for matrix multiplication. [7M]
b) Explain in detail about global common sub expression elimination technique. [7M]

(OR)

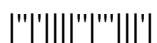
8. a) Discuss in detail the role of dead code elimination and strength reduction during code optimization of a compiler [7M]
b) Draw and explain the Runtime memory organization static storage allocation strategy with pros and cons. [7M]

UNIT-V

9. a) Give an example to show how DAG is used for register allocation. [7M]
b) What are the object code forms? Explain the issues in code generation. [7M]

(OR)

10. a) Discuss about register allocation and assignment in target code generation. [7M]
b) Discuss about how to write a simple code generation algorithm. [7M]



III B. Tech I Semester Regular Examinations, Dec/Jan-2022-23
COMPILER DESIGN

[Common to CSE(AIML), CSE(AI), CSE(DS), CSE(AIDS), AIDS, AIML,CSD]
 Time: 3 hours Max. Marks: 70

Answer any **FIVE** Questions **ONE** Question from **Each unit**

All Questions Carry Equal Marks

UNIT-I

1. a) Discuss about the role of Lexical analyzer, tokens and patterns. [7M]
 b) Explain bootstrapping a compiler with suitable diagrams. [7M]
 (OR)
2. a) How to generate object code for $X=Y+Z*15$ through different phases of compiler? [10M]
 b) Construct a Finite Automaton for the Regular Expression $(00+11)^*010^*10$. [4M]

UNIT-II

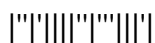
3. a) Verify whether the following grammar is LL(1) or not? [7M]
 $E \rightarrow E + T \mid T$
 $T \rightarrow T^* F \mid F$
 $F \rightarrow (F) \mid a \mid b$.
 b) Construct recursive descent parser for the given grammar. [7M]
 $bexpr \rightarrow bexpr \text{ OR } bterm \mid bterm$
 $bterm \rightarrow bterm \text{ AND } bfactor \mid bfactor$
 $bfactor \rightarrow \text{NOT } bfactor \mid (bexpr) \mid \text{TRUE} \mid \text{FALSE}$
 (OR)
4. a) Explain the rules to perform preprocessing steps of top down parser. Explain with given grammar G . $S \rightarrow Aa \mid bAc \mid Bc \mid bBa$
 $A \rightarrow d \mid B \rightarrow d$. [7M]
 b) Discuss about error recovery strategies in predictive parsing. [7M]

UNIT-III

5. a) Differentiate inherited and synthesized attributes with an example. [7M]
 b) Write Syntax directed definition for constructing syntax tree of an expression derived from the grammar $E \rightarrow E + T \mid E - T \mid T$
 $T \rightarrow (E) \mid id \mid num$. [7M]
 (OR)
6. a) $S \rightarrow L=R \mid R \quad L \rightarrow *R \mid id \quad R \rightarrow L$ construct LALR(1) parser. [10M]
 b) With LALR(1) parse table constructed above check the acceptance of any input string. [4M]

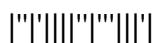
UNIT-IV

7. a) Explain storage allocation strategies used for recursive procedure calls. [7M]
 b) Explain in brief about peephole optimization techniques. [7M]
 (OR)
8. a) What are the principle sources of optimization? Give the classification of code optimization. [7M]
 b) Describe the various fields in an activation record. [7M]



UNIT-V

9. a) Write the algorithm for a simple code generator. [7M]
b) Discuss about register allocation and assignment in target code generation. [7M]
- (OR)
10. a) Explain various issues that affect the efficiency of generated code. [7M]
b) What are the different object code forms in code generation and explain. [7M]



III B. Tech I Semester Regular Examinations, Dec/Jan-2022-23
COMPILER DESIGN

[Common to CSE(AIML), CSE(AI), CSE(DS), CSE(AIDS), AIDS, AIML,CSD]
 Time: 3 hours Max. Marks: 70

Answer any **FIVE** Questions **ONE** Question from **Each unit**

All Questions Carry Equal Marks

UNIT-I

1. a) Write regular expressions for the following languages: Explain operations on Regular expressions. [7M]
 i) All strings over the English alphabet that contain the five vowels in order.
 ii) All strings of a's and b's that do not contain the subsequence abb.
- b) Define lexeme, token and pattern. Identify the lexemes that make up the tokens in the following program segment. Indicate corresponding token and pattern. void swap(int i, int j) { int t; t=i; i=j; j=t; } [7M]

(OR)

2. a) Explain the various phases of a compiler in detail. Also write down the output for the following expression after each phase a: =b*d+20.5. [10M]
- b) Describe the languages denoted by the following regular expressions: [4M]
 (i) $(a|b)^*a(a|b)(a|b)$. (ii) $a^*ba^*ba^*ba^*$

UNIT-II

3. a) Construct the LL(1) parser for G: $S \rightarrow (L) | a$ $L \rightarrow L, S | S$ and check the acceptance of input string (a,(a,a)) [7M]
- b) Compute FIRST and FOLLOW for the grammar: $S \rightarrow S S + \setminus S S * \setminus a$. [7M]

(OR)

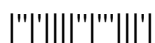
4. a) Eliminate left recursion in the following grammar $A \rightarrow ABd | Aa | a$ $B \rightarrow Be | b$. [7M]
- b) Describe the structure of non recursive predictive parser and error recovery strategies. [7M]

UNIT-III

5. a) Construct the collection of LR(0) item sets and draw the go to graph for the grammar $S \rightarrow S S | a | \epsilon$. Indicate the conflicts (if any) in the various states of the SLR parser. [7M]
- b) Explain in detail how an L-attributed grammar can be converted into a translation scheme. [7M]

(OR)

6. a) Discuss the evolution order of SDTs. Also write its applications [7M]
- b) Write the SDD for a simple type declaration and draw the annotated parse tree for the declaration float a, b, c. [7M]



UNIT-IV

7. a) Explain in brief about Heap Storage allocation strategy. [7M]
b) Explain with suitable example various sources of loop optimization. [7M]

(OR)

8. a) Generate basic blocks and flow graph for quick sort algorithm. [7M]
b) Discuss the role of structure preserving transformations in code optimization. [7M]

UNIT-V

9. a) Give an example to show how DAG is used for register allocation. [7M]
b) What are the object code forms? Explain the issues in code generation. [7M]

(OR)

10. a) Discuss about register allocation and assignment in target code generation. [7M]
b) Discuss about how to write a simple code generation algorithm. [7M]

