



**SIDDHARTH GROUP OF INSTITUTIONS:: PUTTUR
(AUTONOMOUS)**

Siddharth Nagar, Narayanavanam Road – 517583

QUESTION BANK (DESCRIPTIVE)

Subject with Code: Compiler Design (19CS0515)
Year & Sem: III-B.Tech & I-Sem

Course & Branch: B.Tech - CSE
Regulation: R19

**UNIT –I
INTRODUCTION AND LEXICAL ANALYSIS**

1	a	Define Compiler.	[L1][CO1]	[2M]
	b	Explain parts of compiler with neat diagram.	[L2][CO2]	[10M]
2	a	Differences between compiler and Interpreter.	[L4][CO1]	[6M]
	b	Describe about Language Processor in compiler Design?	[L2][CO1]	[6M]
3		Explain the phases of a compiler with neat diagram?	[L2][CO2]	[12M]
4	a	Write in detail about the Structure of Compiler?	[L3][CO1]	[6M]
	b	Analyzethe need for separating lexical analysis and syntax analysis	[L4][CO2]	[6M]
5		Discuss the following terms a) Specification of Tokens b) Recognition of Tokens	[L2][CO1]	[12M]
6	a	Explain in detail about the role of lexical analyzer in Compiler Design.	[L3][CO1]	[6M]
	b	Write about input buffering?	[L3][CO1]	[6M]
7	a	Describe Bootstrapping	[L2][CO1]	[6M]
	b	Summarize Compiler construction Tools	[L5][CO3]	[6M]
8		Design the compiler by using the source program position:=initial+rate*60.	[L6][CO3]	[12M]
9	a	Differentiate tokens, patterns, and lexeme.	[L4][CO1]	[6M]
	b	List the various error recovery strategies for a lexical analysis.	[L1][CO1]	[6M]
10	a	Illustrate Application of compiler technology	[L3][CO1]	[4M]
	b	Explain LEX Tool with the structure of Lex Program?	[L2][CO3]	[8M]

UNIT –II
SYNTAX ANALYSIS AND TOP DOWN PARSING

1	a	Compare left most and right most derivations?	[L4][CO1]	[6M]
	b	Interpret how to eliminate ambiguity for the given Ambiguous Grammar.	[L2][CO1]	[6M]
2	a	Explain the role of parser.	[L2][CO1]	[2M]
	b	Design the recursive decent parser for the following grammar? $E \rightarrow E+T/T$ $T \rightarrow T*F/F$ $F \rightarrow (E)/id$	[L6][CO3]	[10M]
3		Consider the grammar $S \rightarrow AB ABad$ $A \rightarrow d$ $E \rightarrow b$ $D \rightarrow b \epsilon$ $B \rightarrow c$ Construct the predictive parse table and check whether the given grammar is LL(1) or not.	[L6][CO3]	[12M]
4		Consider the grammar $E \rightarrow TE^1$ $E^1 \rightarrow +TE^1 -TE^1 \epsilon$ $T \rightarrow FT^1$ $T^1 \rightarrow *FT^1 /FT^1 \epsilon$ $F \rightarrow (E)/id$ Calculate FIRST and FOLLOW for the above grammar Construct the predictive parse table and check whether the given grammar is LL(1) or not.	[L3][CO2]	[12M]
5	a	Define Ambiguous grammar?	[L1][CO1]	[2M]
	b	Construct Leftmost and Rightmost derivation and parse tree for the string $3*2+5$ from the given grammar. Also check it is ambiguity. Set of alphabets $\Sigma = \{0, \dots, 9, +, *, (,)\}$ $E \rightarrow I$ $E \rightarrow E + E$ $E \rightarrow E * E$ $E \rightarrow (E)$ $I \rightarrow \epsilon 0 1 \dots 9$	[L6][CO2]	[10M]
6	a	Describe the procedure of eliminating Left recursion.	[L1][CO1]	[6M]
	b	Eliminate left recursion for the following grammar $E \rightarrow E+T/T$ $T \rightarrow T*F/F$ $F \rightarrow (E)/id$	[L3][CO1]	[6M]
7	a	Define Context Free Grammar.	[L1][CO1]	[3M]
	b	Show what do you understand by Left factoring. Perform left factor for the grammar $A \rightarrow abB/aB/cdg/cdeB/cdfB$	[L2][CO1]	[9M]
8		Consider the grammar $E \rightarrow E+T/T, T \rightarrow T*F/F, F \rightarrow (E) id$ Construct predictive parsing table and check given grammar is LL(1) or not?	[L6][CO3]	[12M]
9	a	List the types of Parsers available in compiler Design.	[L1][CO2]	[4M]
	b	Explain Error recovery in predictive Parsing.	[L3][CO2]	[8M]
10	a	Illustrate the rules to be followed in the finding the FIRST and FOLLOW.	[L2][CO1]	[6M]
	b	Calculate FIRST and FOLLOW for the following grammar? $E \rightarrow E+T/T$ $T \rightarrow T*F/F$ $F \rightarrow (E)/id$	[L3][CO2]	[6M]

UNIT –III
BOTTOM UP PARSING AND SEMANTIC ANALYSIS

1	a	Describe what is bottom up parsing?	[L1][CO2]	[3M]
	b	Differences between SLR,CLR, LALR parsers?	[L4][CO2]	[9M]
2		Construct CLR Parsing table for the given grammar $S \rightarrow CC$ $C \rightarrow aC/d$	[L6][CO3]	[12M]
3	a	Write about handle pruning?	[L3][CO1]	[4M]
	b	Summarize about SLR parsing?	[L5][CO1]	[8M]
4		Perform Shift Reduce Parsing for the input string using the grammar $S \rightarrow (L)a$ $L \rightarrow L,S S$ a)(a,(a,a)) b)(a,a)	[L6][CO3]	[12M]
5	a	Explain syntax directed definition with simple examples?	[L2][CO2]	[6M]
	b	Describe in detail the Translation scheme of SDD.	[L2][CO2]	[6M]
6	a	Define a syntax-directed translation.	[L1][CO2]	[6M]
	b	Summarise the evaluation order of SDT with an example.	[L5][CO2]	[6M]
7		Discuss Type Checking with suitable examples?	[L2][CO4]	[12M]
8	a	Define augmented grammar?	[L1][CO2]	[2M]
	b	Construct the LR(0) items for the following Grammar? $S \rightarrow L=R$ $S \rightarrow R$ $L \rightarrow *R$ $L \rightarrow id$ $R \rightarrow L$	[L6][CO3]	[10M]
9		Construct the LALR parser for the following Grammar? $S \rightarrow AA$ $A \rightarrow aA/b$	[L6][CO3]	[12M]
10	a	Explain in detail about YACC tool?	[L2][CO3]	[6M]
	b	Describe Synthesized and Inherited attributes with examples.	[L2][CO3]	[6M]

UNIT –IV
RUN TIME ENVIRONMENT AND INTERMEDIATE CODE GENERATION

1	a	Define Activation Record.	[L1][CO5]	[2M]
	b	Sketch the format of Activation Record in stack allocation and explain each field in it.	[L3][CO5]	[10M]
2	a	Discuss about symbol table entries?	[L2][CO4]	[6M]
	b	Write about operations on symbol table?	[L3][CO4]	[6M]
3		Summarise heap management mechanism.	[L5][CO4]	[12M]
4		Describe the Storage Organization with simple examples.	[L2][CO4]	[12M]
5		Define Symbol table. Explain different types of Data structure for symbol table	[L1][CO4]	[12M]
6		Explain Representation of Three Address Codes with suitable Examples	[L2][CO5]	[12M]
7		Produce quadruple, triples and indirect triples for following expression: $(x + y) * (y + z) + (x + y + z)$	[L6][CO5]	[12M]
8		Analyze Different types of Intermediate code with an Example.	[L4][CO5]	[12M]
9	a	Write properties of memory management	[L3][CO4]	[4M]
	b	Discuss Storage allocation strategies with suitable examples?	[L2][CO4]	[8M]
10	a	Describe scope and life time of variable.	[L2][CO4]	[2M]
	b	Illustrate Control Flow Statements.	[L3][CO4]	[10M]

UNIT –V
INTRODUCTION TO APPLICATION LAYER

1		Write about all issues in code generation. Describe it.	[L3][CO5]	[12M]
2	a	Analyze the different forms in target program	[L4][CO6]	[4M]
	b	Explain the target machine architecture?	[L2][CO6]	[8M]
3	a	Discuss the various strategies in register allocation.	[L2][CO4]	[6M]
	b	Describe about loop optimization techniques?	[L2][CO5]	[6M]
4	a	List and explain the Issues in the design of a code generator	[L1][CO6]	[6M]
	b	Define and Show Dead-code elimination with example.	[L3][CO4]	[6M]
5		Interpret about optimization techniques on Basic Blocks with simple examples?	[L2][CO5]	[12M]
6		Explain the peephole optimization Technique with example?	[L6][CO5]	[12M]
7	a	List the applications of DAG.	[L1][CO6]	[4M]
	b	Construct the DAG for following statement. $a+b*c+d+b*c$	[L6][CO6]	[8M]
8		Construct the DAG for the following basic blocks 1. $t1:=4*i$ 2. $t2:=a[t1]$ 3. $t3:=4*i$ 4. $t4:=b[t3]$ 5. $t5:=t2*t4$ 6. $t6:=prod+t5$ 7. $prod:=t6$ 8. $t7:=i+1$ 9. $i:=t7$ if $i \leq 20$ goto 1	[L6][CO6]	[12M]
9	a	Write about Simple code generator	[L3][CO6]	[6M]
	b	Classify Register allocation and register assignment	[L4][CO6]	[6M]
10	a	Describe about global data flow analysis?	[L2][CO6]	[6M]
	b	Discuss function preserving transformations?	[L2][CO6]	[6M]

Preparedby:

Mrs.R.M.Mallika, Mr.B.Pavan Kumar, Mr.P.Krishnamoorthy