



**SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY:: PUTTUR (AUTONOMOUS)**  
Siddharth Nagar, Narayanavanam Road – 517583

**QUESTION BANK (DESCRIPTIVE)**

**Subject with Code:** Compiler Design (18CS0514)

**Course & Branch:** B.Tech- CSIT

**Year & Sem:** III-B.Tech& II-Sem

**Regulation:** R18

**UNIT –I**  
**INTRODUCTION AND LEXICAL ANALYSIS**

<b>1</b>	<b>a</b>	List the various phases of a compiler.	[L1][CO1]	[2M]
	<b>b</b>	Differentiate tokens, patterns, and lexeme.	[L4][CO1]	[2M]
	<b>c</b>	Differences between compiler and Interpreter.	[L4][CO1]	[2M]
	<b>d</b>	Define the Role of Lexical Analyzer.	[L1][CO1]	[2M]
	<b>e</b>	List the various error recovery strategies for a lexical analysis.	[L1][CO1]	[2M]
<b>2</b>		Explain the phases of a compiler with neat diagram?	[L2][CO1]	[10M]
<b>3</b>	a)	Explain in detail about the role of lexical analyzer in Compiler Design.	[L2][CO1]	[5M]
	b)	Write about input buffering?	[L2][CO1]	[5M]
<b>4</b>		Explain about Language Processor in compiler Design?	[L2][CO1]	[10M]
<b>5</b>		Discuss the following terms		
	a)	Specification of Tokens	[L2][CO1]	[5M]
	b)	Recognition of Tokens	[L2][CO1]	[5M]
<b>6</b>	a)	Explain the Structure of Compiler?	[L3][CO1]	[5M]
	b)	What is the need for separating lexical analysis and syntax analysis?	[L2][CO1]	[5M]
<b>7</b>		Explain LEX Tool with a Lex Program?	[L2][CO1]	[10M]
<b>8</b>		Write short notes		
	a)	pass and phases of a compiler	[L3][CO1]	[5M]
	b)	Bootstrapping	[L3][CO1]	[5M]
<b>9</b>		How to design the compiler by using the source program position:=initial+rate*60.	[L1][CO1]	[10M]
<b>10</b>		Write short notes		
	a)	Application of compiler technology	[L3][CO1]	[5M]
	b)	Compiler construction Tools	[L3][CO1]	[5M]

**UNIT –II**  
**SYNTAX ANALYSIS AND TOP DOWN PARSING**

<b>1</b>	<b>a</b>	Define LL(1)?	[L1][CO2]	[2M]
	<b>b</b>	What is Role of Parser?	[L1][CO2]	[2M]
	<b>c</b>	Problems in Top Down Parsing?	[L1][CO2]	[2M]
	<b>d</b>	Define Context Free Grammar.	[L1][CO2]	[2M]
	<b>e</b>	Define Ambiguous grammar?	[L1][CO2]	[2M]
<b>2</b>	a)	Construct the recursive decent parser for the following grammar? $E \rightarrow E+T/T$ $T \rightarrow T * F / F$ $F \rightarrow (E) / id$	[L3][CO2]	[5M]
	b)	Explain about Left factoring and Left Recursion with an examples?	[L2][CO2]	[5M]
<b>3</b>		Calculate FIRST and FOLLOW for the following grammar? a) $E \rightarrow E+T/T$ $T \rightarrow T * F / F$ $F \rightarrow (E) / id$	[L3][CO2]	[5M]
	b)	$S \rightarrow xABC$ $A \rightarrow a bbD$ $B \rightarrow a \epsilon$ $C \rightarrow b \epsilon$ $D \rightarrow c \epsilon$	[L3][CO2]	[5M]
<b>4</b>		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?	[L3][CO2]	[10M]
<b>5</b>	a)	Eliminate left recursion for the following grammar i) $E \rightarrow E+T/T$ ii) $S \rightarrow Aa/b$ $T \rightarrow T * F / F$ $B \rightarrow Bad/c$ $F \rightarrow (E) / id$ $C \rightarrow Cde/f$	[L2][CO2]	[5M]
	b)	Explain about Left factoring with simple example?	[L3][CO2]	[5M]
<b>6</b>		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.	[L3][CO2]	[10M]
<b>7</b>		Consider the grammar $E \rightarrow TE^1$ $E^1 \rightarrow +TE^1   -TE^1   \epsilon$ $T \rightarrow FT^1$ $T^1 \rightarrow *FT^1   / FT^1   \epsilon$ $F \rightarrow GG^1$ $G^1 \rightarrow ^F / \epsilon$ $G \rightarrow (E) / id$ Calculate FIRST and FOLLOW for the above grammar Calculate LL(1) Table for the above grammar.	[L3][CO2]	[10M]
<b>8</b>	a)	Write about left most and right most derivations?	[L3][CO2]	[5M]
	b)	How to eliminate ambiguity for the given Ambiguous Grammar.	[L1][CO2]	[5M]
<b>9</b>		Explain Error recovery in predictive parsing with an Example.	[L1][CO2]	[10M]
<b>10</b>	(a)	Explain parse trees?	[L2][CO2]	[5M]
	(b)	Describe about ambiguity?	[L1][CO2]	[5M]

**UNIT –III**  
**BOTTOM UP PARSING AND SEMANTIC ANALYSIS**

<b>1</b>	<b>a</b>	Define a syntax-directed translation.	[L1][CO3]	[2M]
	<b>b</b>	Differences between SLR, CLR, LALR parsers?	[L2][CO3]	[2M]
	<b>c</b>	Define Handle pruning?	[L1][CO3]	[2M]
	<b>d</b>	What is mean by shift reduce parsing?	[L1][CO3]	[2M]
	<b>e</b>	What is bottom up parsing?	[L1][CO3]	[2M]
<b>2</b>	(a)	Write about handle pruning?	[L3][CO3]	[5M]
	(b)	Write about SLR parsing?	[L3][CO3]	[5M]
<b>3</b>		Construct CLR Parsing table for the given grammar S → CC C → aC/d	[L3][CO3]	[10M]
<b>4</b>		Perform Shift Reduce Parsing for the input string using the grammar. S → (L)a L → L, S   S a)(a, (a, a)) b)(a, a)	[L2][CO3]	[10M]
<b>5</b>		Explain syntax directed definition with simple examples?	[L2][CO3]	[10M]
<b>6</b>		Describe the evaluation order of SDT with an example.	[L2][CO3]	[10M]
<b>7</b>		Discuss Type Checking with suitable examples?	[L2][CO3]	[10M]
<b>8</b>		Explain the Translation scheme of SDD.	[L2][CO3]	[10M]
<b>9</b>		Define augmented grammar? Construct the LR(0) items for the following Grammar? S → L=R S → R L → *R L → id R → L	[L1][CO3]	[10M]
<b>10</b>		Write about YACC tool?	[L3][CO3]	[10M]

**UNIT –IV**  
**RUN TIME ENVIRONMENT AND INTERMEDIATE CODE GENERATION**

<b>1</b>	<b>a</b>	Define Activation Record.	[L1][CO4]	<b>[2M]</b>
	<b>b</b>	Write properties of memory management	[L3][CO4]	<b>[2M]</b>
	<b>c</b>	Describe scope and life time of variable.	[L2][CO4]	<b>[2M]</b>
	<b>d</b>	Define symbol table.	[L1][CO4]	<b>[2M]</b>
	<b>e</b>	Define rules for type checking.	[L1][CO4]	<b>[2M]</b>
<b>2</b>		Draw the format of Activation Record in stack allocation and explain each field in it.	[L4][CO4]	<b>[10M]</b>
<b>3</b>		(a) Discuss about symbol table entries?	[L2][CO4]	<b>[5M]</b>
		(b) Write about operations on symbol table?	[L3][CO4]	<b>[5M]</b>
<b>4</b>		Describe the Storage Organization with simple examples.	[L2][CO4]	<b>[10M]</b>
<b>5</b>		Define Symbol table. Explain different types of Data structure for symbol table	[L1][CO4]	<b>[10M]</b>
<b>6</b>		Write about Different types of Intermediate code with an Example.	[L3][CO4]	<b>[10M]</b>
<b>7</b>		Explain Representation of Three Address Codes with suitable Examples.	[L1][CO4]	<b>[10M]</b>
<b>8</b>		Discuss Storage allocation strategies with suitable examples?	[L2][CO4]	<b>[10M]</b>
<b>9</b>		Write about heap management mechanism.	[L3][CO4]	<b>[10M]</b>
<b>10</b>		Describe about Control Flow Statements.	[L2][CO4]	<b>[10M]</b>

**UNIT –V**  
**CODE OPTIMIZATION AND CODE GENERATION**

<b>1</b>	<b>a</b>	What is the Role of peephole optimization in compilation process	[L1][CO5]	[2M]
	<b>b</b>	List the Issues in the design of a code generator. (any 4)	[L1][CO5]	[2M]
	<b>c</b>	Give the different forms in target program	[L2][CO5]	[2M]
	<b>d</b>	Give the applications of DAG.	[L2][CO5]	[2M]
	<b>e</b>	Define Dead-code elimination with example.	[L1][CO5]	[2M]
<b>2</b>		Write about all issues in code generation. Describe it.	[L3][CO5]	[10M]
<b>3</b>		Explain the target machine architecture?	[L2][CO5]	[10M]
<b>4</b>		Describe about optimization techniques on Basic Blocks with simple examples?	[L2][CO5]	[10M]
<b>5</b>	(a)	Discuss the various strategies in register allocation.	[L2][CO5]	[5M]
	(b)	Write about loop optimization techniques?	[L3][CO5]	[5M]
<b>6</b>		Explain the peephole optimization Technique?	[L2][CO5]	[10M]
<b>7</b>	(a)	Construct the DAG for following statement. $a+b*c+d+b*c$	[L3][CO5]	[5M]
	(b)	Discuss function preserving transformations?	[L2][CO5]	[5M]
<b>8</b>		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. $i:=t7$ 10. if $i \leq 20$ goto 1	[L3][CO5]	[10M]
<b>9</b>		Describe about global data flow analysis?	[L2][CO5]	[10M]
<b>10</b>		Write short notes on		
	i )	Simple code generator	[L3][CO5]	[5M]
	ii)	Register allocation and assignment	[L3][CO5]	[5M]

**Preparedby:**  
**N.SIVA**  
**Assistant Professor/CSE**