GIET MAIN CAMPUS AUTONOMOUS GUNUPUR – 765022

RN190012225

Registration No: | | | | | | | | | | |

Total Number of Pages : 2 AR-17 B.TECH

B.TECH 5th SEMESTER EXAMINATIONS, NOV/DEC 2019

BCSPC5040/BITPC5040 COMPILER DESIGN

Common to CSE/IT Branch

Time : 3 Hours Maximum : 100 Marks

Answer ALL Questions

The figures in the right hand margin indicate marks.

PART – A: (Multiple Choice Questions) 10 x 2=20 Mark

Q.1. Answer *All* Questions

a	The output Lexical Analyzer is a) Lexeme b) Tokens c) Pattern d) None of the above	[CO1] [PO1]
b	The grammar G={ S->aSb/ab } generates a)a^nb^n where n>0 b) a^nb^m where n>0 c) a^mb^n where n>0 d) None of the above	[CO1] [PO1]
c	First of A in the Grammar G={S->aS/bA/e, A->eS/Se/f} a) e,f b) e,f,a c) e,f,a,b d) a,b	[CO1] [PO1]
d	The following strategy follows backtracking a)Recursive Descent Parser b) Brute force Method c)LL(1) Parser d) None	[CO2] [PO1]
e	LALR parsing table uses the following a) LR(0) items b) LR(1) items c) LR(2) items d) LR(3) items	[CO2] [PO1]
f	The following is not one of the intermediate forms. a) Three Address Code b)Polish Notation c) Triple d) Infix	[CO3] [PO1]
g	Static Allocation supports a) Recursion b) Dynamic Memory to some extent c) Runtime Memory Modification d) None	[CO3] [PO1]
h	What is the Prerequisite for Code Optimization? a) Preserving Syntax b) Semantics should be preserved c) Space Occupied must be minimum d) The execution speed must be less.	[CO4] [PO1]
i	The following techniques operate on compiler time a) Copy Propagation b) Constant Folding c)Strength Reduction d) Common Sub Expression Elimination	[CO4] [PO1,2]
j	Compiler optimization where expensive operations are replaced with equivalent but less expensive operations a) Strength Reduction b) Dead Code Elimination c) Loop Unrolling d) Constant Folding	[CO4] [PO1]

PART – B: (Short Answer Questions) 10X2=20 Marks

Q.2. Answer *ALL* questions

a	Outline the differences between NFA and DFA.	[CO1] [PO1]
b	Summarize the advantage of LEX tool.	[CO1] [PO1]
c	Demonstrate when the grammar can be ambiguous.	[CO2] [PO1]
d	Compare and contrast top down and bottom up parser.	[CO2] [PO1]
e	Define synthesized attribute.	[CO2] [PO1]
f	List the contents of Activation Record	[CO3] [PO1]
g	Write any three memory allocations by the compiler.	[CO3] [PO1]
h	Outline the significance of copy propagation.	[CO4] [PO1]
i	Summarize the importance of Loop Fusion.	[CO4] [PO1]
j	Demonstrate the need of constant folding.	[CO4] [PO1]

PART – C: (Long Answer Questions) 4X15=60 Marks

Answer *ALL* questions

Q.3			
a	Compute the First and Follow for the following grammar. E -> TE' E' -> +T E'|ϵ T -> F T' T' -> *F T' | ϵ F -> (E) | id	7	[CO1] [PO1,2]
b	Summarize the need of left factoring for top down parsers.	8	[CO1] [PO1,2]
	OR		
c	Design LL(1) parser for the following grammar. S->aBA/d/ eA ; B->eB/f/e ; A->SB/d/e	7	[CO1] [PO1,2]
d	Design recursive descent parser for the grammar S->aS/ABa/e , A->eA/d, B->e.	8	[CO1] [PO1,2]
Q.4			
a	Design CLR parsing table for the following grammar. S → Ab ; A → aA ; A → b	15	[CO2] [PO1,2,3]
	OR		
b	Design LALR parsing table for the following grammar. S->CC; C->cC/d	15	[CO2] [PO1,2]
Q.5			
a	Develop quadruples, triples and indirect triples for the expression: -a + a * (b + c) + (b + c) * d .	8	[CO3] [PO 1,2,3]
b	What is dependency graph? How it is different from parse tree? Explain in brief with an example.	7	[CO3] [PO1]
	OR		
c	Develop quadruples, triples and indirect triples for the expression: (a+b) * (c+d) * (a+b+c)	7	[CO3] [PO1]
d	Distinguish L-attributed from S-attributed grammars.	8	[CO3] [PO1]
Q.6			
a	Discuss the importance of common sub expression elimination.	7	[CO4] [PO1]
b	Discuss various Loop optimization techniques with examples.	8	[CO4] [PO1]
	OR		
c	Discuss various machine independent code optimization techniques in detail.	7	[CO4] [PO1]
d	Summarize the basics of flow of control optimization.	8	[CO4] [PO1]

==0==