



**CVR COLLEGE OF ENGINEERING**  
*UGC Autonomous Institution* - Affiliated to JNTUH

<b>R18</b>
------------

**B. Tech III Year I Sem. Main & Suppl. Exams Feb - 2022 (2019 & 2018 Batches)**

**Subject: Automata and Compiler Design**

**Branch: IT & CSIT**

**Time: 3 hours**

**Max. Marks: 70**

**Note:**

1. Please verify the regulation of question paper and subject name.
2. Assume required data, if not given in the question.

**Answer Any Five Questions (5 X 14 = 70 Marks)**  
**All Questions Carry Equal Marks**

1. Outline various phases of compilation and give the output for each phase of compilation considering the expression  $C = A + B * 20.0$  BL2 (CO1)
2. a) Illustrate shift reduce parsing by considering the CFG  $S \rightarrow CC, C \rightarrow cC \mid d$  and  $w = cdd$ . BL3 (CO2)  
 b) Demonstrate the role of handle pruning in bottom up parsing. BL3 [7+7]
3. Identify the importance of symbol table and enumerate various storage allocation strategies. BL3 (CO3)
4. Discuss how intermediate code generation helps to achieve code optimization. BL5 (CO4)
5. a) Develop the process of constructing basic block from a flow graph and demonstrate its generation from three address code statements by identifying basic blocks. BL3 (CO5)  
 b) Examine how peephole optimization eliminates redundant common sub expressions. BL4 [7+7]
6. a) Construct  $\epsilon$  – NFA for the given regular expression  $(a+b)^*abb$ . BL3 (CO1)  
 b) Construct NFA to accept strings  $abb, aba, bab$ . Hence obtain equivalent DFA for the NFA. BL3 [7+7]
7. a) Explain the SDT scheme for type checking. BL5 (CO3)  
 b) Compare and contrast between different languages based on Chomsky hierarchy. BL2 [7+7]
8. a) Build the Straight forward code generation algorithm to create the object code for the computation of  $x = (a+b) - (e-(c+d))$ . BL3(CO5)  
 b) Inspect how order of three address code can reduce the cost of computation by applying the concept of DAG to compute  $x = (a+b)-(e-(c+d))$ . BL4 [7+7]