

Reg. No. :

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|

Question Paper Code: 31261

B.E. / B.Tech. DEGREE EXAMINATION, MAY 2017

Sixth Semester

Computer Science and Engineering

01UCS601 - PRINCIPLES OF COMPILER DESIGN

(Regulation 2013)

Duration: Three hours

Maximum: 100 Marks

Answer ALL Questions

PART A - (10 x 2 = 20 Marks)

1. Define Lexemes, Tokens, and Patterns.
2. Distinguish between compiler and interpreter.
3. Differentiate between final states in a NFA and a DFA.
4. What is the use of an error handler?
5. Write a brief note on YACC.
6. Write the drawbacks of shift-reduce parser.
7. Illustrate why every S-attributed definition is L-attributed.
8. What is annotated parse tree?
9. List the applications of DAG.
10. What is peephole optimization?

PART - B (5 x 16 = 80 Marks)

11. (a) With a neat sketch, discuss the phases of a compiler.

(16)

Or

- (b) What are the phases of compiler? Explain with a neat diagram. Also write down the output for the following expression after each phase $a: = b + c * d / e$. (16)
12. (a) (i) Construct an NFA to recognize the regular expression $(a^+ b | a^+)^+$, Obtain its equivalent DFA and minimize the number of states in DFA. (10)
- (ii) How is finite automata useful for lexical analysis? (6)

Or

- (b) Design a Lexical analyzer generator. Also write the sample code which includes declaration, translation rules and auxiliary procedures. (16)
13. (a) Consider the following context free grammar $G = (\{S, A, B\}, S, \{a, b\}, P)$ where P is
- $S \rightarrow Aa / bAc / dc / bda$
- $A \rightarrow d$.
- Show that this grammar is LALR (1) but not SLR (1). (16)

Or

- (b) Consider the following grammar and construct a SLR parsing table for the same
- $E \rightarrow E + T$
- $T \rightarrow T * F$
- $T \rightarrow F$
- $F \rightarrow (E)$
- $F \rightarrow id$ (16)
14. (a) Write the syntax directed definition for generating 3-address code for an assignment statement. (16)

Or

- (b) With supporting examples, describe the run time storage management. (16)
15. (a) (i) Elaborate the issues involved in design of a code generator. (8)
- (ii) Explain looping and flow graphs. (8)

Or

- (b) Explain the code optimization techniques using examples. (16)