

Reg. No. :

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Question Paper Code : 71689

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

Sixth Semester

Computer Science and Engineering

CS 6660 — COMPILER DESIGN

(Common to Information Technology)

(Regulations 2013)

Time : Three hours

Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. Define the two parts of compilation.
2. List the cousins of the compiler?
3. Write a regular expression for an identifier and number.
4. What are the various parts in LEX program?
5. Eliminate the left recursion for the grammar $A \rightarrow Ac \mid Aad \mid bd$.
6. What are the various conflicts that occur during shift reduce parsing?
7. What do you mean by binding of names?
8. Mention the rules for type checking.
9. What is a basic block?
10. What do you mean by copy propagation?

PART B — (5 × 16 = 80 marks)

11. (a) What are the phases of the compiler? Explain the phases in detail. Write down the output of each phase for the expression $a := b + c * 60$. (16)

Or

- (b) (i) Explain briefly about compiler construction tools. (6)
(ii) Describe in detail about Cousins of compiler? (4)
(iii) Draw the transition diagram for relational operators and unsigned numbers. (6)
12. (a) Convert the Regular Expression $abb(a|b)^*$ to DFA using direct method and minimize it. (16)

Or

- (b) (i) Differentiate between lexeme, token and pattern. (6)
(ii) What are the issues in lexical analysis? (4)
(iii) Draw the transition diagram for relational operators and unsigned numbers. (6)

13. (a) Construct a predictive parsing table for the grammar

$$S \rightarrow (L) | a$$

$$L \rightarrow L, S | S.$$

and show whether the following string will be accepted or not. $(a, (a, (a, a)))$. (16)

Or

- (b) Consider the following Grammar

$$E \rightarrow E+T | T$$

$$T \rightarrow TF | F$$

$$F \rightarrow F^* | a | b$$

Construct the SLR parsing table for the above grammar. (16)

14. (a) What are the different storage allocation strategies? (16)

Or

- (b) (i) Explain in detail about Specification of a simple type checker (10)
(ii) Explain about the parameter passing. (6)

15. (a) Discuss the various issues in design of Code Generator. (16)

Or

(b) (i) Explain in detail about optimization of Basic Blocks. (8)

(ii) Construct the DAG for the following Basic Block. (8)

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).
-