CSE 323 (R-15)

Total No. of Questions :09] [Total No. of Pages : 02

III/IV B.Tech. DEGREE EXAMINATIONS, NOVEMBER- 2019

Second Semester

COMPUTER SCIENCE ENGINEERING

COMPILER DESIGN

Time: Three Hours **Maximum marks:60**

Answer Question No.1 Compulsory **6X2=12 M**

Answer ONE Question from each Unit **4X12=48 M**

1. a) Write the regular definition for arithmetic expressions

b) Give an example to eliminate the left recursion with rules

c) Brief on error recovery in LR parser

d) Discuss the evaluation of semantic rules

e) Mention various parameter passing mechanisms

f) Give the organization of optimizing compiler.

UNIT-I

2. Explain in detail the process of compilation. Illustrate the output of each phase of compilation of the input “a=(b+c)*(b+c)*2”

(OR)

3. a) Define the following terms: compiler, interpreter, Translator and differentiate between them.

b) Differentiate between lexeme, token and pattern. What are the issues in lexical analysis? Explain.

UNIT-II

4. a) Why lexical and syntax analyzer are separated out? Explain.

b) Construct the predictive parser for the following grammar.

S->(L) | a L ->L,S | S

P.T.O

(OR)

5. a) Show that the grammar S -> 0S1 |SS| $\in$ is ambiguous.

   b) Explain the Non-Recursive predictive parsing with an example.

   c) What is left recursion and left factoring?

UNIT-III

6. a) Give the Syntax Directed Translation scheme for desk calculator.

   b) Write short notes on (i) Formats of three address code (ii) Construction syntax tree for expressions.

(OR)

7. a) What is meant by activation of procedure? How it can be represented with activation tree and record? Explain with quick sort example.

   b) Compare bottom up approaches of parsing with all top down approaches.

   c) Write the quadruple, triple for the expression: -(a*b) + (c+d)- (a+b+c+d)

UNIT-IV

8. a) Can we reuse the symbol table space? Explain through an example.

   b) Write the algorithm to generate basic blocks and flow graph for quick sort algorithm.

(OR)

9. a) Explain about the sources and criterions of code optimization as machine dependent and independent types.

   b) Explain Symbol table organization techniques.

CSE 323 (R-15)

Total No. of Questions :09] [Total No. of Pages : 02

III/IV B.Tech. DEGREE EXAMINATIONS, APRIL- 2019

Second Semester

COMPUTER SCIENCE ENGINEERING

COMPILER DESIGN

Time: Three Hours **Maximum marks:60**

Answer Question No.1 Compulsory 6X2=12 M

Answer ONE Question from each Unit 4X12=48 M

1. a) Write a LEX program to identify comments in the program

   b) Goals of error handler in a parser?

   c) Differentiate synthesis and inherited translation

   d) Describe the structure of entries in symbol table

   e) Compare deep access and shallow access

   f) What is instruction scheduling?

UNIT-I

2. What are the various phases of the compiler? Explain each phase in detail. Elaborate on compiler construction tools.

(OR)

3. a) Describe the various phases of compiler and trace the program segment 4: * + = c b a for all phases.

   b) Define CFG and explain its capabilities.

UNIT-II

4. Give the classification of parsing techniques and briefly explain each.

(OR)

5. a) What are the limitations of recursive descent parser?

   b) Discuss how Brute-Force approach operates in top down parsing.

   c) Construct SLR parsing table for the following grammar:

   S -> L = R, S -> R, L -> *R, L -> id, R-> L

P.T.O

UNIT-III

6. a) What is an intermediate code? Explain different types of intermediate codes forms and represent the following statement in different forms:

   W=(A+B) - (C+D) + (A+B+C)

   b) Explain how to generate three address codes with syntax directed definitions with an example.

(OR)

7. a) Write the quadruple, triple, indirect triple for the expression:

   (a*b) + (c+d)- (a+b+c+d)

   b) For the following code generate three-address code.

```
if (a < b+c*20)
{
        a=a*b - 50
        d=(a/b) + 25;
        print (a,d)
}
```

UNIT-IV

8. a) Write about the techniques in local and global transformations.

   b) What is scope of variable? Write about various ways to access non local variables.

(OR)

9. Write the algorithm to generate basic blocks and flow graph for quick sort algorithm and apply code optimization techniques on flow graph generated for quick sort.