C

Reg. No. : | | | | | | | | | | |

Question Paper Code: 95C01

B.E. / B.Tech. DEGREE EXAMINATION, APRIL 2024

Fifth Semester

Computer Science and Business Systems

19UCB501 - Compiler Design

(Regulations 2019)

Duration: Three hours Maximum: 100 Marks

Answer ALL Questions

PART A - (5 x 1 = 5 Marks)

1. Compiler should report the presence of _____ in the source program, in translation process. CO1- U

   (a) Classes (b) Objects (c) Errors (d) Text

2. Which of the following is a top down parser? **CO1- U**

   (a) recursive descent parser (b) shift reduce parser

   (c) operator precedence parser (d) SLR parser

3. _____ is the activity of filling up unspecified information of labels using appropriate semantic actions during the code generation process. CO1- U

   (a) SDT (b) Back patching (c) Function Call (d) Recursive Call

4. How many points can we define for the basic block with 8 statements? CO1- U

   (a) 9 (b) 8 (c) 7 (d) 10

5. Code generator uses _____ function to determine the status of available registers and the location of name values. CO1- U

   (a) setReg (b) cinReg (c) pfReg (d) getReg

PART – B (5 x 3= 15 Marks)

6. Illustrate the language processing system. CO1- U

7. Consider the following grammar E→E+E| E * E| (E) | id obtain Left Most Derivation of the string, id+id+id. CO2- App

8. What are the different representations of intermediate code? CO1- U

9. What is peephole and what is the need of peephole optimization? CO1- U

10. What are the types of storage allocation strategies? CO2- App

PART – C (5 x 16= 80 Marks)

11. (a) Illustrate how the following high level language statement is transformed into machine code during the compilation process **x=(a+b) * (c+d)** with the neat sketch of phases of compiler. CO2- App (16)

Or

(b) Obtain DFA for the regular expression (l(l/d)*. CO2- App (16)

12. (a) Design Predictive parser for the Grammar CO2- App (16)

S->(L) |a

L->L,S|S

and parse the input string (a,a) .

Or

(b) Check the following grammar is SLR(1) or not? CO2- App (16)

S->L=R|R

L->*R | id

R->L

13. (a) Explain in detail the various representation of intermediate code. CO1- U (16)

Or

(b) Explain the translation of Arithmetic expressions in detail. CO1- U (16)

14. (a) Explain the principal sources of optimization in detail. CO1- U (16)

Or

(b) Describe peephole optimization with necessary examples CO1- U (16)

15. (a) For the statement x = a / (b + c) – d*(e + f), generate three address c and subsequent target code using the simple code generation algorithm. CO2- App (16)

Or

(b) Develop a quick sort algorithm for reads nine integers into an array and sorts them by using the concepts of activation tree. CO2- App (16)