M.C.A. 1st Semester (MCA 2 Year Programme)
w.e.f. 2020-2021 Examination, November–2023
COMPILER DESIGN
Paper–20MCA21C2

Time allowed : 3 hours]   *[Maximum marks : 80*

Note: Attempt *five* questions in all. *Question No.1 is compulsory.* In addition to compulsory question, attempt *four* more questions selecting *one* question from each unit.

1. **Compulsory question:**

   (a) What do you mean by system programming? Explain the components of system programming.

   (b) What do you understand by cross compiler?

   (c) What are the two types of conflicts in shift reduce parsing? Give example.

   (d) Define parsing. Classify the types of parsing.

   (e) What is hashing? Discuss.

   (f) Differentiate between Abstract Syntax tree and DAG representation of intermediate code.

   (g) List out different object code forms.

   (h) What is code optimization? Illustrate with example.

67005-N-P-4-Q-9 (23)   [P.T.O.

Unit-I

2. (a) Explain the problems faced by a one-pass assembler. Draw and explain the detailed flowchart for pass-2 of a two-pass assembler.

    (b) What are different loading schemes? Explain absolute loader scheme with its advantages and disadvantages.

3. (a) What are the basic functions of loaders? Differentiate absolute, relative and bootstrap loader.

    (b) State the basic tasks a macro instruction processor performs. Explain how the nested macro calls are executed with example?

Unit-II

4. (a) Differentiate between:
        (i) Passes and phases of compiler
        (ii) Syntax analysis and semantic analysis

    (b) Construct the canonical LR(l) item sets for the following grammar:

    S ⟶ AA

    A ⟶ aA/b

5. (a) Compare and contrast SLR with LALR. Define Kernel items and Non-kernel items.

67005-N

Show the following grammar is LALR(l)

s-> Aa | bAc | de | bda

A-> d

(b) What are the problems with top down parsing? Write the algorithm to remove left recursion from a grammar with example.

Unit-III

6. (a) Explain quadruples and triples with example. Write three address code for the expression:

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

(b) With a neat diagram explain the format of the Symbol Table. And discuss the tree structures representation of scope information.

7. (a) Explain various data structure used for implementing symbol table and compare them.

(b) What are different intermediate code forms? Discuss different Three Address code types and implementations of Three Address statements.

Unit-IV

8. (a) Explain the main issues of code generation in detail.

(b) Define peephole optimization. List the characteristics of peephole optimization.

[P. T. O.

9.   (a)   Explain DAG representation of basic blocks with example.

     (b)   Discuss the following code optimization techniques with examples:

           (a)   Constant propagation

           (b)   Strength reduction

           (c)   Code Motion

67005-N

MCA 1st Semester (MCA 2 Year Programme) w.e.f. 2020-21 Examination – December, 2024

COMPILER DESIGN

Paper : 20MCA21C2

Time : Three hours]   *[Maximum Marks : 80*

Before answering the questions, candidates should ensure that they have been supplied the correct and complete question paper. No complaint in this regard, will be entertained after examination.

Note :   Attempt *five* questions in all, selecting *one* question from each Unit. Question No. 1 is *compulsory*. All questions carry equal marks.

1. *Compulsory question* :

   (a) What is bootstrapping ?

   (b) What do you understand by cross compiler ?

   (c) What are the *two* types of conflicts in shift reduce parsing ? Give example.

   (d) Define left recursion. Is the following grammar left recursive : E->E+E | E*E | a | b ?

(e) What is hashing ? Discuss.

(f) Differentiate between Abstract Syntax tree and DAG representation of intermediate code.

(g) Explain the algorithm for basic block identification.

(h) What is code optimization ? Illustrate with example.

UNIT – I

2. (a) What is system software ? Explain working of system software tool. Give any *five* example of system software tools.

(b) What is loader ? Differentiate between direct linkage and reallocating loader scheme with an example.

3. (a) Explain any *four* components available as integral part of system programming environment.

(b) What is self-relocating program ? Explain with appropriate example.

UNIT – II

4. (a) Describe the mathematical model used for creating finite automata.

(b) What functions are performed by code Optimizer and code generator during synthesis phase ? Discuss.

5. (a) Consider the production :

S→aAb

A→cd/C

Show that recursive descent parsing fails for input string "acdb", also explain Recursive Descent Algorithm.

(b) Why it is important to perform operator precedence during compiler processing.

UNIT – III

6. (a) Give the general structure of activation record. Explain the purpose of each component.

(b) What do you mean by attributed grammars ? Discuss the translation scheme for Converting an infix expression to its equivalent postfix form.

7. (a) Explain the use of symbol table in compilation process. List out the various attributes for implementing the symbol table.

(b) Describe how three address code can be represented as triples.

UNIT – IV

8. (a) Explain various code optimization techniques. Discuss the strategies for loop optimization and dead code elimination.

(b) How register allocation and assignment is performed during code generation ? Explain.

9. (a) Explain the different storage allocation strategies.

(b) What are major sources of optimization in a given code ? Discuss.

———————

(d) What is Finite Automata ?

(e) Write a short note on Peephole optimization.

(f) What do you mean by a cross-compiler ?

(g) Explain error handling in LR parsing.

(h) What is Subroutine linkage ?

Unit-I

2. (a) What are linkers and loaders ? Explain static and dynamic scope for linking. Also, differentiate between compiler and assembler.

(b) What do you mean by System Software Tools ? Explain the variety of software tools in detail.

3. Explain the following in detail :

(a) General Loader Schemes

(b) Direct Linkage Loader

Unit-II

4. (a) Define Compiler. How is it different from interpreter ? Explain various phases of a compiler.

(b) Define bottom-up approach of parsing. What are its advantages ? Discuss various LR parsers.

5. (a) Explain the role of lexical analysis. Describe various lexical-errors. Also write a brief note on regular expressions.

(b) Define top-down parsing. Also explain the problems associated with top-down parser.

Unit-III

6. (a) What is meant by syntax directed definition ? Explain the order of syntax directed definitions in detail.

(b) Define symbol table. What are various data structures that are used for a symbol table ?

7. Explain the following in detail :

(a) Intermediate code generation

(b) Syntax Trees and Construction of syntax trees

Unit-IV

8. (a) What do you mean by Code generation ? Explain the various issues in code generation in detail.

(b) What do you mean by code optimization ? Explain the principal sources of optimization in detail.

9. Explain the following in detail :

(a) Design of a simple code Generator

(b) Global Data Flow Analysis

Roll No. : ....

Total No. of Questions : 9]　　　[Total No. of Pages : 4

67005-N

MCA 1st Semester (Regular)
Examination, March-2022
(MCA 2 Year Programme)
(w.e.f. 2020-21)
Paper-20MCA21C2
COMPILER DESIGN

Time : **Three Hours**]　　　[Maximum Marks : **80**

Before answering the questions, candidates should ensure that they have been supplied the correct and complete question paper. No complaint in this regard, will be entertained after examination.

Note :– Attempt *five* questions in all, selecting *one* question from each Unit. Q. No. 1 is compulsory. All questions carry equal marks.

1. (a) Explain Go Loader.

(b) What is Loop Optimization ?

(c) Explain the role of Parser.