

2014

Time : 3 hours

Full Marks : 80

Candidates are required to give their answers in their own words as far as practicable.

The figures in the margin indicate full marks.

Answer from both the Groups as directed.

Group – A

(Objective Type Questions)

1. Choose the correct alternative of the following : 2×10 = 20

   (a) Right side of three address code has how many numbers of operand at most ?

   (i) 3

   (ii) 2

(iii) 4

(iv) 5

(b) A grammar that produces more than one parse tree for some sentence is said to be :

(i) Ambiguous

(ii) Context free

(iii) Disambiguous

(iv) Regular

(c) What is not the phase of a compiler ?

(i) Syntax analyzer

(ii) Code generator

(iii) Code optimizer

(iv) Code linker

(d) YACC builds up :

(i) SLR parsing table

(ii) Canonical LR parsing table

(iii) LALR parsing table

(iv) None of these

(e) An annotated parse tree is :

(i) A parse tree with attribute values shown at the parse tree nodes.

(ii) A parse tree with values of only some attributes shown at parse tree nodes

(iii) A parse tree without attribute values shown at parse tree nodes

(iv) A parse tree with grammar symbols shown at parse tree nodes

(f) Consider the statement "fi(x>=10)", where 'if' has been misspelled. The error is detected by the compiler in the phase :

(i) Lexical analysis

(ii) Syntax analysis

(iii) Semantic analysis

(iv) Syntactic analysis

(g) The role of predictive parsing is :

(i) To construct a top-down parser that never backtracks

(ii) To construct a top-down parser that backtracks

(iii) To construct a bottom-up parser that never backtracks

(iv) To construct a bottom-up parser that backtracks

(h) Type checking is normally done during :

(i) Syntax analysis

(ii) Lexical analysis

(iii) Code optimization

(iv) Syntax directed translation

(i) Which of the following is true regarding LL(1) grammar ?

(i) No ambiguous or left recursive grammar can be LL(1)

(ii) No ambiguous but left recursive grammar can be LL(1)

(iii) No left recursive but ambiguous grammar can be LL(1)

(iv) Both ambiguous and left recursive grammar can be LL (1)

(j) Which of the following statement is true in the context of left factoring ?

(i) It produces output suitable for recursive decent parsing

(ii) It produces output suitable for predictive parsing

(iii) It produces output suitable for top-down parsing

(iv) Both (i) and (ii)

Group – B

(Long-answer Type Questions)

Answer any **four** questions :

2. How many phases a complier normally consists of ? Illustrate each phase with suitable example. 15

3. By taking a suitable example, illustrate the

working of a top down predictive parsing technique. 15

4. What is a symbol table ? Why is it needed in complier design ? Explain different components a symbol table with example. 15

5. What is intermediate codes in compilers ? Why is it needed in compiler design ? Discuss different types of intermediate codes generated by intermediate code generation phase. 15

6. What is DAG ? What is the use of DAG in complier construction ? Construct a DAG for the following statement. 15

$$Z = X - Y + X * Y * U - V / W + X + V$$

7. Generate a parse tree for expression a + b * c based on grammar for arithmetic expression. 15

8. Construct DFA for ab* | ba* and minimized the number of states of the above constructed DFA. 15

9. Write short notes on any **three** of the following : 5×3 = 15

(a) Data activation record

(b) Peephole optimization

(c) Error recovery strategies in complier

(d) Symbol table

(e) Handling Reserved Keywords