2017

CSIT

PAPER: 4.2./ 506 (Old)

COMPILER DESIGN

MARKS:80

TIME : 3 hours

(Question 1is Compulsory and Answer any Five rest of the questions)

(The figures in the margin indicate full marks for the questions)

1.Answer the following: 1X5=5

(i) Bottom-up parsing is a technique to find—

(a) Leftmost derivation

(b) Rightmost derivation

(c) Leftmost derivation in reverse

(d) Rightmost derivation in reverse

(ii) A compiler that runs on one machine and produces the target code for another machine is known as--------

(a)cross compiler

(b)Linker

(c) Preprocessor

(d)Assambler

(iii) A group of logically related characters in the source program is known as---

(a) Token
(b) Lexeme
(c) Parse Tree
(d) Buffer

(iv)Which of the following is the least powerful parsing method ?

(a) LL(1) (b) Cannonical LR

(c) SLR (d)LALR

(v) Which two functions are required to construct a parsing table in predictive parsing technique?

(a) CLOSURE ( ) and GOTO ( )

(b) FIRST ( ) and FOLLOW ( )

(c) ACTION ( ) and GOTO ( )

(d) All of the above

2.(a)Construct a DFA for language over alphabet $\Sigma=\{a,b\}$ that will accept all string ending with '**ab**'. 3

(b) $M=(\{q_1,q_2,q_3\},\{0,1\},\delta,q_1,\{q_3\})$ is NDFA where δ is given by

$\delta(q_1,0)=\{q_2,q_3\}$ $\delta(q_1,1)=\{q_1\}$

$\delta(q_2,0)=\{q_1,q_2\}$ $\delta(q_2,1)=\phi$

$\delta(q_3,0)=\{q_2\}$ $\delta(q_3,1)=\{q_1,q_2\}$

Construct an equivalent DFA. 8

(c) Convert the given Regular Expression into DFA 4 (a/b)* a (a/b)

3.(a) What is ambigious grammar? Prove the grammar is ambigious

E→E+E / E*E / (CE) / id

Explain how ambiguity can be removed. 1+3+5=9

(b) Eliminate Left recursion for the following grammar: 3

S→S+E / E

S→ E*F / F

F→(S) / id

(c) Perform Left factoring for the following grammar: 3

A→ aBcC / aBb / aB / a

B→ ε

C→ε

4.(a) What is LEX? Write structure of Lexprograme. 3+4=7

(b) Using parsing table Show that the given grammar is not LL(1) . 8

E→E+T/T

T→T*F/F

F→(E)/id

5. What is Top-down and Bottom-up parsing ? Define handle and handle pruning.Explain shift-reduce parsing with stack implementation. 5+4+6=15

P.T.O.

6.(a) Explain LR parser with different component. 7

(b) Construct the LR(0) items for the following grammar: 8

$E \rightarrow E+T/T$

$T \rightarrow T*F/F$

$F \rightarrow (E)/id$

7.Write short notes (Any Three): 3X5=15

(a) LL(1) grammar

(b) Regular Expression

(c) Role of Parser

(d) Cross -compiler

(e) Predictive Parser

****