CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY

Seventh Semester of B. Tech. Examination (C.E.)
March-2018
CE401 Compiler Construction (CC)

Date: 28.03.2018, Wednesday Time: 10:00 a.m. To 01:00 p.m. Maximum Marks: 70

Instructions:

1. The question paper comprises of two sections.
2. Section I and II must be attempted in separate answer sheets.
3. Make suitable assumptions and draw neat figures wherever required.
4. Rough work is to be done in the last page of main supplementary, please don't write anything on the question paper.
5. Indicate clearly, the option(s) you attempt along with its respective question no.
6. Figures to the right indicate full marks.

SECTION-I

Q-1 **Answer the following questions.**

1. Explain the analysis synthesis model of compilation. List the factors that affect the design of a compiler. Also List major functions done by compiler. **5**
2. Construct a DFA for a given regular expression (010+00)*(10)* **4**
3. How do the parser and scanner communicate? Explain with the block diagram communication between them. **2**

Q-2

[A] Compare Inherited Attributes with Synthesized Attributes. **2**

[B] Construct predictive parsing table for following. **6**

S -> A
A -> aB | Ad
B -> bBC | f
C -> g

OR

[B] Explain various code optimization techniques. **6**

[C] Explain: Error Recovery Strategies in Compiler in brief. **4**

Q-3

[A] Enlist and discuss the challenges in the Code Generation phase of the Compiler specifically. **4**

[B] Describe the LEX file structure with an example. **4**

OR

[B] Which are the design considerations for constructing a compiler? Which are the major issues in designing (constructing) the compiler? **4**

[C] Find out FIRST & FOLLOW sets for all the Non-terminals **4**

S→ AcB | cbB | Ba
A→ da | BC
B→ g | Є
C→ h | Є

OR

[C] Construct predictive parsing Table for the following grammar. **4**

E→S+T | S-T

T→ V | V*V | V/V

V→ a | b.

Q-4

1. What are the features of a good language? What are the features of a good Compiler? **4**
2. Show that both the regular expressions (a|b)* and (a*|b*)* are equivalent. **3**
3. Consider the CFG: S → SS+ | SS* | a
   Show that the string aa+a*can be generated by this grammar. Construct a parse tree and syntax tree for this string. **4**

Q-5

[A] Construct RDP for the following grammar. (assume suitable string) **4**

E → TE'

E'→ +T E' |A

T → FT

T'→*FT' | A

F →(E) | id

[B] Prepare DAG for: Sum = Sum + ((a/b)*c)/(a/b) **4**

OR

[B] Show that the following grammar is **LR(1)** but not LALR(1): **4**

S→Aa | bAc | Bc | bBa

A→ d

B→ d

[C] Construct the **canonical parsing table** for the following grammar: **4**

S'→S

S→CC

C→cC| d

OR

[C] What is meant by activation record? List out the fields in the activation record. **4**

Q-6

[A] Parse the string abbcde using S/R technique for the following Grammar: **4**

S→aABe

A→Abc| B

B→d

OR

[A] Compute the operator precedence relation graph and precedence relation table for given grammar. **4**

G→ E

E→ E+T | E-T

T→ T*F | T/F

F → id

[B] Write Syntax Directed Definition for translating following grammar for postfix notation. Also draw annotated parse tree for 9-5+2. **4**

expr → expr + term
expr → expr – term
expr→ term
term → 0 | 1 |…..|9

OR

[B] Compare: Static v/s Dynamic Memory Allocation **4**

[C] What is Intermediate Form of the code? Which are the advantages of it? What are generally used intermediate forms? **Write N-Tuple notation for**: (a+b)*(c+d)-(a+b+c) **4**

OR

[C] Prepare the SLR parsing Table with error detection for the following grammar. **4**
E→ E+E | E * E | (E) | id