53 (CS 601) CPDG

2019

COMPILER DESIGN

Paper : CS 601

Full Marks : 100

Time : Three hours

The figures in the margin indicate full marks for the questions.

*Answer **any five** questions.*

1. Consider the following instruction :
$d = a + b/50$
and discuss the different phases of compiler. 20

2. 5+10+5=20

   (a) Design an NFA, accepts the regular expression $(X/Y)^* XY$.

   (b) Create an equivalent DFA.

   (c) Minimize the number of states in your DFA (if possible).

Contd.

3. 5+15=20

(a) Write down the rules for computing FIRST and FOLLOW of a grammar.

(b) Use them for the grammar mentioned below :

$S \rightarrow AaBbD,\ A \rightarrow aA \mid \in$

$B \rightarrow b,\ D \rightarrow d \mid \in$

4. 5+10+5=20

(a) Remove left recursion for the following grammar :

$A \rightarrow A+B \mid B,\ B \rightarrow B*C \mid C$

$C \rightarrow id.$

Consider your new grammar for the next question.

(b) Create the LR (0) items.

(c) Check whether $id + id*id$ will be accepted or not.

5. 10+10=20

(a) Create LR (1) items for the grammar mentioned below :

$S \rightarrow AB,\ A \rightarrow a,\ B \rightarrow bB \mid d$

(b) Using the LR(1) items, create an LALR(1) parsing table.

6. 10+10=20

Consider the following program segment :

```
void function 1 (              )
{ int a[100], LB, UB
      LB=0
      UB=100
      while (LB ≤ UB)
      {     a[LB]=10
            LB=LB+1
      }
}
```

(a) Write down three-address codes for the above mentioned program segment.

(b) Use Local and Global common subexpression elimination techniques to optimize your code.

7. Write short notes on : 4×5=20

(a) Left Factoring

(b) Augmented Grammar

(c) DAG

(d) Tokens.