USN | | | | | | | | | | |

10IS662

Sixth Semester B.E. Degree Examination, Dec.2017/Jan.2018

Compiler Design

Time: 3 hrs. Max. Marks:100

Note: ***Answer any FIVE full questions, selecting at least TWO questions from each part.***

PART – A

1 a. Explain input buffering strategy used in lexical analysis phase. **(10 Marks)**

b. Write a transition diagram for recognizing unsigned numbers. Sketch the program segment to implement it, showing the first/initial state and one final state. **(10 Marks)**

2 a. What is recursive-descent parser? Trace and explain the working of the recursive-descent parser for the input "bcd" and grammar:

$A \rightarrow bCd$

$C \rightarrow ce|e$ **(10 Marks)**

b. $S \rightarrow (L)|a$

$L \rightarrow L, S|S$

Make the grammar suitable for top down parsing. Construct predictive parse table and parse the string (( )). **(10 Marks)**

3 a. Show that the following grammar is LL(1) grammar, without constructing any parse table. Also construct first and follow set for the given grammar.

$S \rightarrow AaAb|BbBa$

$A \rightarrow \in$

$B \rightarrow \in$ **(06 Marks)**

b. Explain "handle" and "handle pruning". Show the handles during the parse of input "$id_1 * id_2$" for the grammar.

$E \rightarrow E + T|T$

$T \rightarrow T*F|F$

$F \rightarrow id|(E)$ **(06 Marks)**

c. What are the different conflicts encountered during shift-reduce parsing? Explain the conflicts during the parse of the input "id*id" for the grammar given in Q3(b). **(08 Marks)**

4 a. For the given grammar, construct SLR(1) parse table and parse the string "aa".

$S \rightarrow SA|A$

$A \rightarrow a$ **(10 Marks)**

b. Construct canonical parse table for the grammar:

$S \rightarrow CC$

$C \rightarrow cC|d$ **(10 Marks)**

PART – B

5 a. Write an SDD for simple desktop calculator. Show the annotated parse free for the expression (5*7) + (1*2). **(10 Marks)**

b. Explain parser stack implementation of postfix SDT with an example. **(10 Marks)**

6 a. Explain the following with example:
   i) Value membered method for constructing DAG
   ii) Triples
   iii) Indirect triples
   iv) Quadruples
   v) Static-single assignment form **(15 Marks)**

   b. Write an algorithm for unification of a pair of nodes in a type graph. **(05 Marks)**

7 a. Describe the structure of activation record. Also explain the task division between a caller and callee in implementing the procedure calls. **(12 Marks)**

   b. Explain the design goals of a garbage collector. **(08 Marks)**

8 a. Write the algorithm for partitioning three-address instructions into basic blocks. Generate intermediate code for the following statements and identify the basic blocks, (given w = 8 bytes).

```
for i from 0 to 10 do
    for j from 0 to 10 do
        Result = c[i, j] * d[i, j];
for i from 0 to 09 do
        a[i, i] = 1.0;
```

(10 Marks)

   b. Find liveliness and next-use info for the following code block. Given only c, d, e are live on exit.
   i) d = b * c
   ii) e = a + b
   iii) b = b * c
   iv) a = e – d
   Construct DAG and simplify the above code. **(10 Marks)**

* * * * *