Sixth Semester B.E. Degree Examination, December 2010
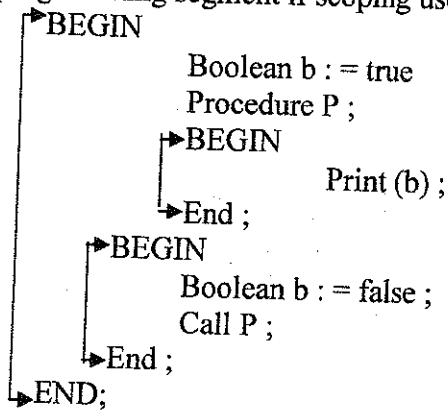## Compiler Design

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 with neat diagram, the various phases of a compiler. Mention the input and output for each phase. **(08 Marks)**

   b. Define static and dynamic scoping. Explain the working and output of the following programming segment if scoping used is static and dynamic:

```
BEGIN
        Boolean b : = true
        Procedure P ;
        BEGIN
                Print (b) ;
        End ;
    BEGIN
        Boolean b : = false ;
        Call P ;
    End ;
END;
```
   (04 Marks)

   c. With an example, explain the use and coordination between 'LEX' and 'YACC' the compiler writing tools. **(08 Marks)**

2 Consider the grammar:

$$E \rightarrow 5 + T | 3 - T$$

$$T \rightarrow V | V * V | V + V$$

$$V \rightarrow a | b$$

   a. What is the use of left factoring? Do the left factoring for the above grammar. **(04 Marks)**

   b. Write an algorithm to obtain the FIRST and Follow table. Obtain FIRST and Follow table for the above grammar. **(08 Marks)**

   c. Write an algorithm to construct the predictive parsing table. Construct predictive parsing table for the above grammar. **(08 Marks)**

3 Consider the grammar:

$$S \rightarrow E\#$$
$$E \rightarrow E - T$$
$$E \rightarrow T$$
$$T \rightarrow F \uparrow T$$
$$T \rightarrow F$$
$$F \rightarrow (E)$$
$$F \rightarrow i$$

   a. Write the algorithm to construct basic finite state control m/c for SLR (1) and action α goto functions entries. **(08 Marks)**

   b. Construct the following for the above grammar:
      i) Basic finite state control.
      ii) SLR (1) parsing table containing action and goto function entries. **(12 Marks)**

4   Consider the grammar:

$G \rightarrow S$

$S \rightarrow E = E$

$S \rightarrow f$

$E \rightarrow T$

$E \rightarrow E + T$

$T \rightarrow f$

$T \rightarrow T * f$

when terminal symbols are $\{=,+,*,f\}$

   a.   Write an algorithm to construct finite state control for LR(1) parser.   **(08 Marks)**

   b.   Construct LR(1) finite state control and explain the algorithm to construct parsing table containing action α goto function entries.   **(12 Marks)**

PART – B

5   a.   With an example, explain the concept of syntax directed definition.   **(08 Marks)**

   b.   Write the grammar and syntax directed definitions for a simple desk calculator and show annotated parse tree for the expression (3+4)*(5+6).   **(12 Marks)**

6   a.   What is DAG? Construct a DAG for the following expression, $a + a * (b - c) + (b - c) * d$.
   (04 Marks)

   b.   With an example, explain the various formats of intermediate code.   **(10 Marks)**

   c.   Write quadruple representation for, $a + a * (b - c) + (b - c) * d$.   **(06 Marks)**

7   a.   Explain the run time storage scheme for C++-language. Give the structure of activation record and explain with suitable example.   **(12 Marks)**

   b.   Explain the design goals for garbage collectors.   **(08 Marks)**

8   a.   Discuss the following terms:

     i)   Basic blocks

     ii)   Next-use information

     iii)   Flow graph   **(10 Marks)**

   b.   Explain the following code optimization with example:

     i)   Finding local common sub expression.

     ii)   Dead code elimination.   **(10 Marks)**

* * * * *