# Sixth Semester B.Tech. Degree Examination, May 2016
## (2013 Scheme)
## 13.602 : PRINCIPLES OF PROGRAMMING LANGUAGES (R)

Time : 3 Hours

Max. Marks : 100

## PART – A

Answer **all** questions.

1. Distinguish between static scoping and dynamic scoping with an example.

2. What is the difference between normal-order and applicative-order evaluation ? What is lazy evaluation ?

3. Explain the significance of the parameter in object-oriented languages.

4. Describe the tuple and set types of Python.

5. What is a Just-In-Time (JIT) compiler ? What are its potential advantages over interpretation or conventional compilation ?                    **(5×4=20 Marks)**

## PART – B

Answer **any one full** question from **each** Module. **Each full** question carries **20** marks.

## Module – I

6. a) Discuss about structured sequence control statements with suitable examples.    12

   b) Consider the following more complex declarations.

   var r : record

   x : integer;

   y : char;

   A : array [1..10, 10..20] of record

   z : real;

   B : array [0..71] of char;

P.T.O.

end ;

end ;

var j, k : integer;

Assume that these declarations are local to the current subroutine. Note the lower bounds on indices in A; the first element is A[1, 10]. Describe how r would be laid out in memory. Then show code to load r.A[2, j].B[k] into a register. Be sure to indicate which portions of the address calculation could be performed at compile time.                                                        8

OR

7. a) Explain the difference between row-major and column-major layout for contiguously allocated arrays. Why does a programmer need to know which layout the compiler uses ? Why do most language designers consider row-major layout to be better ?                                                        12

   b) Describe the jump code implementation of the short-circuit Boolean evaluation.     8

### Module – II

8. a) Describe how to maintain the static chain during a subroutine call.                   8

   b) In your favorite language with generics, write code for simple versions of the following abstractions :

   a) a stack, implemented as a linked list

   b) a priority queue, implemented as a skip list or a partially ordered tree embedded in an array

   c) a dictionary (mapping), implemented as a hash table.                              12

OR

9. a) Describe the Prolog search strategy. Discuss backtracking and the instantiation of variables.                                                          14

   b) Describe three alternative means of allocating coroutine stacks. What are their relative strengths and weaknesses ?                                          6

## Module – III

10. a) What happens to the implementation of a class if we redefine a data member ? For example, suppose we have :

    class foo
    {public : int a;
    char *b; };
    ...
    class bar : public foo
    {public : float c;
    int b; };

    Does the representation of a bar object contain one b field or two ? If two, are both accessible, or only one ? Under what circumstances ? **10**

    b) What are the characteristics of scripting languages ? List the principal ways in which scripting languages differ from conventional "systems" languages ? **10**

    OR

11. a) Explain the importance of virtual methods for object closures. **8**

    b) Compare the approaches to object orientation taken by Perl 5, PHP 5, JavaScript, Python and Ruby. **12**

## Module – IV

12. What is a semaphore ? What operations does it support ? How are they used to implement synchronization ? How do binary and general semaphores differ ? **20**

    OR

13. a) Briefly describe three different implementation strategies for a symbolic debugger. **6**

    b) Summarize the synchronization mechanisms of Ada 95, Java, and C#. Compare and contrast them with one another. **14**

———————