

THAPER INSTITUTE OF ENGINEERING AND TECHNOLOGY

Department of Mechanical Engineering

End Semester Examination (December 13<sup>th</sup>, 2006)

66

Subject Code: CD-004

Max Marks: 60

Subject: Computer Programming and Data Structures

Max Time: 3 Hour

Class: M.E. CAD-CAM and Robotics Engineering

Instructor: Mr. R K Duvedi

Note: 1. Attempt any five questions.

2. Answers to all subsections of a question must be given in order at one location.

3. The evaluated answer sheets can be seen on 14<sup>th</sup> December, 2006 (Thursday), in the office of course instructor at 4.00 p.m.

- 1(a) What are the different types of *instructions* used in C/C++. 3
- 1(b) What do you mean by *Escape Sequences*? Explain the use of the various *Escape Sequences* used in C/C++. 3
- 1(c) Briefly explain the various types of *Constants* and *Variables* used in C, along with their format specifiers. 3
- 1(d) Explain with the help of suitable example and flow chart the working of *for* Loop. 3
- 2 Explain the use of following with suitable examples.
  - (i) Logical Operators 2
  - (ii) Function Overloading 2
  - (iii) Inline Function 2
  - (iv) Return by Reference 3
  - (v) Multiple Inheritance 3
- 3(a) Differentiate within the following with the small examples
  - (i) *break* and *goto* statement 2
  - (ii) *return* and *exit()* 2
  - (iii) Passing argument *by value* and *by reference* 2
- 3(b) What is a *Pointer Variable* and *Pointer Constant*? Explain the use of the *void* Pointer. Also briefly explain the various valid pointer arithmetic operations. 6
- 4 Write a computer program in C/C++, having the a structure Vect, which has two member variables x and y, of type float to represent the X and Y coordinates of a point (or vector). Overload the following operators to carry out the following operations: 12
  - (i) '+' to add two variables of type Vect.
  - (ii) '-' to find the subtraction of two vectors
  - (iii) '\*\*' to find the product of a vector and a floating point variables
  - (iv) '\*\*' to find the Dot product of the two vectors

Also make the following functions:

(i) *Display()* to print any n number of elements of an array of type Vect.

(ii) *Get\_Vect()* to get data for any n number of elements from the user to be stored in an array of type Vect.

In function *main()*, make two arrays of type Vect to hold 10 elements each. Make use of function *Get\_vect()* to get the elements of both the arrays. Make use of a switch-case statement, where in different cases make use of various overloaded operators to operate upon the elements of the two arrays made earlier (without changing the values in the original arrays), and make use of function *Display()* to display the results in different cases.

- 5      Make a class *String*, having only one private member variable *Str* which is a pointer to string. make the suitable *constructors* for initializing the member variables of the objects of the class *String*. Also make the following public member functions: 12
- (i) *Get\_string()* to obtain the string from the user having one argument of type pointer to the string
  - (ii) *reverseit()* to reverse the order of the alphabets of a string passed as an argument and assign the reversed string to the member variable *Str* of the object of the class *String* (using swap operation).
  - (iii) *Display()* to display the contents of the object of the class *string*.
  - (iv) Overload the operator '+' to add two strings and store the concatenated string in the member variable of the object of class *String*.

In *main()* initialize two objects of *S1* and *S2* of class *String*, reverse the string *S1* and then add string *S1* and *S2* and store the concatenated string in object *S3*. Finally display the contents of all strings.

- 6(a)    What are the various storage classes in C/C++. Explain with respect to their initialization, visibility and life time, with suitable examples. 6
- 6(b)    Differentiate between *publically* and *privately* derived classes. Thus explain member function overriding with respect to derived classes. 6
- 7(a)    Make a Linked list to store the information of the objects of a class *Student*, having *Name*, *Roll\_No*, and *CGPA* as the member variables of suitable data type. Make suitable constructors for class *Student* along with two public member functions *Get\_Data()* and *Print\_Data()*. 8
- Make the suitable functions for :
- (1) Getting the header of the linked list.
  - (2) Adding an element to the linked list at any position P.
  - (3) Deleting a link from the list of student from any position P.
  - (4) Display the entire List.
- In *main()*, make the linked list *L1*, and make use of an infinite loop to keep on working on the list until user wants to continue, and switch-case statement for carrying out all the operations on the string.
- 7(b)    What is a *binary tree*? Briefly describe any three methods of traversing a binary tree. 4