**CPP Problem Design**

|  |
| --- |
| **Subject:** **Class Point in plane** |
| **Contributor: 陳俊儒, 林承達, 廖宣瑋** |
| **Main testing concept:** Class member function   |  |  | | --- | --- | | **Basics** | **Functions** | | ■ C++ BASICS  □ FLOW OF CONTROL  ■ FUNCTION BASICS  □ PARAMETERS AND OVERLOADING  □ ARRAYS  ■ STRUCTURES AND CLASSES  □ CONSTRUCTORS AND OTHER TOOLS  □ OPERATOR OVERLOADING, FRIENDS,AND REFERENCES  □ STRINGS  □ POINTERS AND DYNAMIC ARRAYS | □ SEPARATE COMPILATION AND NAMESPACES  □ STREAMS AND FILE I/O  □ RECURSION  □ INHERITANCE  □ POLYMORPHISM AND VIRTUAL FUNCTIONS  □ TEMPLATES  □ LINKED DATA STRUCTURES  □ EXCEPTION HANDLING  □ STANDARD TEMPLATE LIBRARY  □ PATTERNS AND UML | |
| **Description:**  The type Point is a fairly simple data type, but under another name (the template class pair) this data type is defined and used in the C++ Standard Template Library, although you need not know anything about the Standard Template Library to do this exercise. Write a definition of a class named **Point** that might be used to store and manipulate the location of a point in the plane. You will need to declare and implement the following member functions:  a.A member function set that sets the private data after an object of this class is created.  b. A member function to move the point by an amount along the vertical and horizontal directions specified by the first and second arguments.  c.A member function to rotate the point by 90 degrees clockwise around the origin.  d.Two const inspector functions to retrieve the current coordinates of the point.  Document these functions with appropriate comments. Embed your class in a test program that requests data for several points from the user, creates the points, then exercises the member functions.  **Input:**  Replace the main from main.cpp.  **Output:**  Please see sample out.  **Sample Input / Output：**   |  |  | | --- | --- | | Sample Input | Sample Output | | main.in | 0 5  1 7  7 -1  -1 -7  -7 1  1 7 | |
| **□ Eazy,Only basic programming syntax and structure are required.**  **■ Medium,Multiple programming grammars and structures are required.**  **□ Hard,Need to use multiple program structures or more complex data types.** |
| **Expected solving time:**  20minutes |
| **Other notes:** |