**CPP Problem Design Example**

|  |
| --- |
| **Subject: Prime Number** |
| **Contributor: 張子樂，廖宣瑋 ，謝公耀** |
| **Main testing concept: Class 設計**   |  |  | | --- | --- | | **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:**  Define a class named **PrimeNumber** that stores a prime number.   * The class PrimeNumber has only one variable **value(int)**. * The class PrimeNumber has two constructors: * **PrimeNumber()**: construct a PrimeNumber where the **value** is 1. * **PrimeNumber(int \_value)**: construct a PrimeNumber where the **value** is \_value. * You should implement the following function: * **get()**: return the value of this PrimeNumber. * And you are required to Overload all the following operators: * **++**: return the next larger prime number. * **--**: return the next smaller prime number.   \*\* If the PrimeNumber value equals to 2, -- operator should return 1. In the testing data, the PrimeNumber won’t be less than 2.  **Input:**  The main() function in your submission will be replaced when judging.  You can use the main() function in “Other Notes” to test your program.  No inputs for this exercise.  **Output:**  The result of executing your program with the given main function.  **Sample Input / Output：**   |  |  |  | | --- | --- | --- | |  | **Sample Input** | **Sample Output** | | 第一組測資與輸出 | No inputs | 2  2  13  17 | | … |  |  | |
| ■易，僅需用到基礎程式設計語法與結構  □中，需用到多項程式設計語法與結構  □難，需用到多項程式結構或較為複雜之資料型態或結構 |
| **Expected solving time:**  15分鐘 |
| **Other notes:**  int main()  {  PrimeNumber p1, p2(13);  PrimeNumber a = ++p1;  PrimeNumber b = p2++;  cout << a.get() << endl;  cout << p1.get() << endl;  cout << b.get() << endl;  cout << p2.get() << endl;  system("pause");  return 0;  } |