2015/05/14

Program Design
Second Mid-Test – 2015

I. Matching: choose the correct fittings (20 %)

______1. Iterator　　______2. friend function
______3. Data abstraction　　______4. static class variable
______5. this pointer　　______6. Pointer-based arrays
______7. Dangling pointer　　______8. Memberwise assignment
______9. class string　　______10. Copy constructor

a) Used when only one copy of a variable should be shared by all instances of a class.
b) Describing functionality of a class independent of its implementation.
c) An object that "walks through" a collection.
d) Defined outside the class's scope, yet has access to private members of the class.
e) An implicit argument to all non-static member-function calls.
f) A constructor that takes as its argument a reference to an object of the same class as the one in which the constructor is defined.
g) The default behavior of the = operator.
h) Problem that may occur when a pointed object is out of the scope.
i) Provides member function substr.
j) Do not provide range-checking.

II. Closing: finishing the sentence (15 %)

a) Class members are accessed via the ______ operator in conjunction with the name of an object (or reference to an object) of the class or via the ______ operator in conjunction with a pointer to an object of the class.
b) ______ must be used to initialize **constant** members of a class.
c) A nonmember or global function must be declared as a(n) ______ of a class to have access to that class's private data members.
d) An object's non-static member functions have access to a "self pointer" to the object called the ______ pointer.

III. Another representation of the time for the Time class is to store *the number of total seconds since midnight* rather than the integer values of hour, minute and second. Clients could use the same public methods and get the same results. Implement the following Time class without changing the interface of the class. (40%)

```
class Time {
 public:
     Time( int = 0, int = 0. int = 0 ); // default constructor

     void setTime( int. int. int ); // set hour, minute. second
     void setHour( int ): // set hour
     void setMinute( int ): // set minute
     void setSecond( int ): // set second

     int getHour(); // return hour
     int getMinute(); // return minute
     int getSecond(); // return second

     void printUniversal(); // print universal time
     void printStandard(); // print standard time
 private:
     int totalSeconds; // number of seconds since midnight
}; // end class Time
```

a) Define a constructor that takes three integers *h, m, s* as arguments and uses ***member initializer syntax*** to initialize data member **totalSeconds**. (hint h* 3600+m*60+s) (10%)
b) On the other hand, define a constructor that takes three integers *h, m, s* as arguments and calls ***setTime*** member function to create a Time object. (5%)
c) Define the ***setTime*** member function that takes three integers *h, m, s* as arguments and calls ***setHour(h)***, ***setMinutes(m)***, and ***setSecond(s)*** member functions to initialize a Time object. (5%)
d) Define the ***setHour*** member functions that takes integers *h* as argument and calls ***getMinute*** and ***getSecond*** to calculate the correct values for the **totalSeconds** data member. (5%)
e) If in the main() scope, you create a constant **Time** object as **const Time noon(12,0,0),** how do you modify the definitions of the get functions and the two print functions in order to get data members and print the time of the **noon** object? (5%)
f) For an object ***t*** able to perform the following operation: **t.setHour(18).setMinute(0).setSecond(0)** and **t.setTime(18,0,0).printstandard()** how do you modify the definitions of these member functions? (use **setHour(int h)** as example. (10%)

IV. Complex numbers have the form: (25%, 5,10,10)

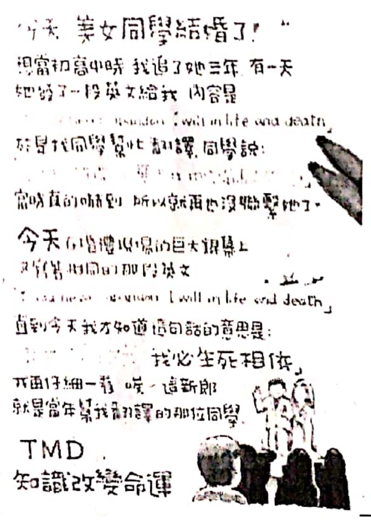
realPart + imaginaryPart * *i*

where *i* has the value $\sqrt{-1}$

a) Please create a C structure ***Complex*** with two double variable fields for the **realPart** and **imaginaryPart.**
b) Please create a class ***ComplexClass*** that has a data member of ***Complex*** structure. Define a constructor that accept two arguments, e.g. 3.2, 7.5 to initialize the data member. Make this constructor a default constructor by assigning both the **realPart** and the **imaginaryPart** to values 1.0. The constructor also prints out a message like:

Complex number (3.2+7.5i) is constructed.

c) Define an **add** member function that accepts two ***Complex*** structures, adds them together, and returns the result as a ***Complex*** structure.
d) 今天你笑了嗎？



「今天 美女同學結婚了！」
想當初高中時 我追了她三年 有一天
她寫了一段英文給我 內容是
「[illegible] abandon I will in life and death」
於是找同學幫忙翻譯，同學說：
「[illegible]」
當時我的嚇到 所以就再也沒聯繫她了。
今天 在婚禮現場的巨大銀幕上
又看到當時的那段英文
「[illegible] abandon I will in life and death」
直到今天我才知道這句話的意思是：
「[illegible] 我必生死相依」
我再仔細一看 唉，這新郎
就是當年幫我翻譯的那位同學。
TMD。
知識改變命運