**C++程式設計HW 4 (Chapter 7)**

**Part 1, Choice(s) 選擇題 (35%)**

1) In an ADT the implementation details are \_\_\_\_\_\_\_\_ the interface through which a program uses it.

A) a part of

B) determined by

C) kept separate from

D) used by

E) None of the above

2) Objects are created from ADTs that encapsulate data and \_\_\_\_\_\_\_\_ together.

A) constants

B) functions

C) memory addresses

D) variables

E) None of the above

3) Objects permit data hiding. This means they can keep functions outside the class from \_\_\_\_\_\_\_\_ the data.

A) accessing

B) changing

C) knowing the names of the variables holding

D) deleting

E) doing all of the above

4) When three different objects of a particular class are created, they are said to be separate \_\_\_\_\_\_\_\_ of the class.

A) members

B) ADTs

C) instances

D) children

E) None of the above

5) In OOP terminology, an object's member variables are often called its attributes \_, and its member functions are sometimes referred to as its methods.

A) values, operators

B) data, activities

C) attributes, activities

D) attributes, methods

E) values, activities

6) When the body of a member function is defined inside a class declaration, it is called a(an) \_\_\_\_\_\_\_\_ function.

A) static

B) global

C) inline

D) conditional

E) constructor

7) A \_\_\_\_\_\_\_\_ is a member function that is automatically called when a class object is \_\_\_\_\_\_\_\_.

A) constructor, created

B) constructor, destroyed

C) destructor, created

D) destructor, destroyed

E) both A and D

8) A constructor may have a return type of \_\_\_\_\_\_\_\_.

A) int

B) bool

C) void

D) Any of the above

E) None of the above

9) \_\_\_\_\_\_\_\_ member function may be called by a statement in a function that is outside the class.

A) A declared

B) A public

C) A private

D) An inline

E) Any

10) A C++ member function that uses, but does not change, the value of a member variable is called a(n) \_\_\_\_\_\_\_\_.

A) accessor

B) mutator

C) user

D) constant

E) constructor

11) Accessors are sometimes called \_\_\_\_\_\_\_\_ functions and mutators are sometimes called \_\_\_\_\_\_\_\_ functions.

A) set, get

B) get, set

C) public, private

D) private, public

E) regular, inline

12) If Circle is the name of a class, which of the following statements would create a Circle object named myCircle?

A) myCircle Circle;

B) myCircleCircle();

C) Circle myCircle;

D) Circle myCircle();

E) None of the above

13) If setRadius is a Circle class function and myCircle is a Circle object, which of the following statements would set myCircle's radius to 2.5?

A) setRadius(2.5);

B) myCircle.setRadius(2.5);

C) Circle.setRadius(2.5);

D) Circle(setRadius(2.5));

E) None of the above

14) When an object is passed \_\_\_\_\_\_\_\_ to a function, its members are not copied.

A) as an argument

B) by value

C) by reference

D) as a constant reference

E) by either method C or D above

15) Class declarations are usually stored \_\_\_\_\_\_\_\_.

A) on CDs

B) in their own header files

C) in .cpp files, along with function definitions

D) under pseudonyms

E) in system files

16) The bundling of an object's data and procedures together is called \_\_\_\_\_\_\_\_*.*

A) OOP

B) encapsulation

C) data hiding

D) structuring

E) private access

17) If you do not declare an access specification, the default for members of a class is \_\_\_\_\_\_\_\_.

A) inline

B) public

C) private

D) global

E) shared

18) The \_\_\_\_\_\_\_\_ is used to protect important data.

A) public access specifier

B) private access specifier

C) protect() member function

D) class protection operator

E) default constructor

19) Public members of a class object can be accessed from outside the class by using the \_\_\_\_\_\_\_\_.

A) dot operator

B) get function

C) extraction operator

D) member access operator

E) class name

20) A C++ member function that sets or changes the value stored in a member variable is called a(n) \_\_\_\_\_\_\_\_.

A) accessor

B) mutator

C) user

D) getter

E) updater

21) A(n) \_\_\_\_\_\_\_\_ member function may only be called from a function that is a member of the same class.

A) public

B) private

C) overloaded

D) local

E) constructor

22) A constructor must have the same name as the \_\_\_\_\_\_\_\_.

A) first private data member

B) first public data member

C) class

D) first object of the class

E) function return type

23) The name of a destructor must begin with \_\_\_\_\_\_\_\_.

A) the name of the class

B) a tilde (~)

C) a capital letter

D) an underscore

E) None of the above

24) A class may have \_\_\_\_\_\_\_\_ default constructor(s) and \_\_\_\_\_\_\_\_ destructor(s).

A) only one, only one

B) only one, more than one

C) more than one, only one

D) more than one, more than one

E) no, only one

25) The \_\_\_\_\_\_\_\_ directive prevents a header file from being included in a program more than once.

A) #include

B) #define

C) #ifndef

D) #endif

E) #exclude

26) When a member function is defined outside of the class declaration, the function name must be qualified with the class name, followed by \_\_\_\_\_\_\_\_.

A) a semicolon(;)

B) the scope resolution operator (::)

C) the public access specifier

D) the private access specifier

E) a tilde (~)

27) A class can have a member variable that is an instance of another class. This is called \_\_\_\_\_\_\_\_.

A) object composition

B) object containment

C) chaining

D) encapsulation

E) None of the above

28) A structure variable is similar to a class object in which of the following way(s)?

A) It has member data that is usually private and accessed through public member functions.

B) Its data can be initialized with a constructor.

C) It can be passed to a function or returned from a function.

D) All of the above.

E) B and C, but not A.

29) When an object or structure variable is passed to a function as a constant reference \_\_\_\_\_\_\_\_.

A) the function accesses the original object, rather than a copy of it

B) the function cannot make any changes to the member variables

C) it is more efficient than passing it by value

D) All of the above are true

E) A and B are true, but not C

30) The process of object-oriented analysis includes the following key steps \_\_\_\_\_\_\_\_.

A) create a problem description, find all the verbs in the description, then create the classes

B) identify the needed data members and member functions, then assign a class name

C) identify the private and public variables, then prototype the functions and write the code

D) write the main function, then determine which classes it will use

E) identify the needed classes, define their attributes and behaviors, and identify relationships between classes

31) Which of the following is(are) true?

A) A data type consists of a collection of values together with a set of basic operations defined on these values. Take int for example, the int data type has certain specified values, such as 0,1,-1,2,…, and the operations for the type int consists of +, -, \*, /, %, and a few other operators and predefined library functions.

B) A data type is called abstract data type (ADT) if the programmers who use the type do not have access to the details of how the values and operations are implemented.

C) The programmer defined type: classes and the C++ predefined types, such as int are ADTs.

D) A programmer who uses a class should not need to even look at the definitions of member functions, but need to know how the data of the class is implemented.

E) The terms: information hiding, data abstraction, and encapsulation are the most common used terms which mean that the details of the implementation of a class are hidden from the programmer who uses the class

32) Which of the following are legal access to the class or struct members? Assume each is outside of the class member definitions,

struct S class C class D

{ { {

int x; int x; public:

int y; int y; int x;

} private: int y;

S s; int z; private:

}; int z;

C c; };

D d;

A) s.x

B) c.x

C) d.x

D) c.z

E) d.z

33) For the three classes declared as follows:

class A class B class C

{ { {

public: public: public:

void f(); B(); C(int k,int m);

private: private: C();

int x; int y; int g();

}; int z; private:

}; int y,z;

};

Which of the following is(are) legal?

1. B b1();
2. C c1(3,4);
3. C c2(3);
4. A a1;
5. C c3;.
6. Given the program, which of the following class member accesses are legal?

#include <iostream>using namespace std;class Automobile{public: void setPrice(double p); void setProfit(double p);// other public members private: double price; double profit; double getProfit();

// other private members

};int main(){

Automobile c1,c2;

double aPrice,bProfit;

c1.price = 9999.99; // A)

c2.setPrice(19999.99); // B)

cout<<(aPrice=c1.getPrice()); // C)

bProfit = c2.getProfit(); // D)

if(c1.profit == c2.profit) // E)

cout<< "Happy\n";

}

35) When you defined a C++ class, which of the following should not be part of the interface?

1. all declarations of private member variables
2. all declarations for public member functions
3. all explanatory comments for public member declarations.
4. all declarations for private member functions
5. all member function definitions (public or private)

**Part 2, True/False 是非題 (5%)**

1) True/False: It is legal to call a constructor as a member function of an object of a class, as in

class A

{

public:

A(){ }

A(int x, int y):xx(x), yy(y) { }

// other members

private:

int xx;

int yy;

};

int main()

{

A w;

w.A(2,3); // **Is this legal?**

}

2) True/False: A structure has member variables, like an object, but they are usually all public and accessed directly with the dot operator, instead of by calling member functions. //public by default

3) True/False: Inline functions are always more efficient than noninline functions.//pass by ref

4) True/False: Consider these hierarchical structures.

struct Date

{

int year;

//members

};

struct Person

{

Date birthDay;

//other members

};

Person Bill;

The year of Bill’s birthday may be accessed as Bill.year;

5) True/False: When an object is defined without an argument list for its constructor, the compiler automatically calls the object's default constructor, if there is one.

**Part 3, Coding 程式題(60%)**

1. **(10%) Employee**

Write a class named Employee that has the following member variables:

* name. A string that holds the employee’s name.
* idNumber. An int variable that holds the employee’s ID number.
* department. A string that holds the name of the department where the employee works.
* position. A string that holds the employee’s job title.

The class should have the following constructors:

* A constructor that accepts the following values as arguments and assigns them to the appropriate member variables: employee’s name, employee’s ID number, department, and position.
* A constructor that accepts the following values as arguments and assigns them to the appropriate member variables: employee’s name and ID number. The department and position fields should be assigned an empty string ( "" ).
* A default constructor that assigns empty strings ( "" ) to the name , department , and position member variables, and 0 to the idNumber member variable.

Write appropriate mutator functions that store values in these member variables and accessor functions that return the values in these member variables. Once you have written the class, write a separate program that creates three Employee objects to hold the following data.

Name ID Number Department Position

Susan Meyers 47899 Accounting Vice President

Mark Jones 39119 IT Programmer

Joy Rogers 81774 Manufacturing Engineer

The program should store this data in the three objects and then display the data for each employee on the screen.

1. **(10%) Inventory & Cash Register Class**

First design an Inventory class that can hold information for an item in a retail store’s inventory.

The class should have the following private member variables.

|  |  |
| --- | --- |
| **Variable Name** | **Description** |
| itemNumber | An int that holds the item’s number. |
| quantity | An int that holds the quantity of the item on hand. |
| cost | A double that holds the wholesale per-unit cost of the item |

The class should have the following public member functions.

|  |  |
| --- | --- |
| **Member Function** | **Description** |
| default constructor | Sets all the member variables to 0. |
| constructor #2 | Accepts an item’s number, quantity, and cost as arguments. Calls other class functions to copy these values into the appropriate member variables. Then calls the setTotalCost function. |
| setItemNumber | Accepts an intargument and copies it into the itemNumbermember variable |
| setQuantity | Accepts an intargument and copies it into the quantity member variable. |
| setCost | Accepts a double argument and copies it into the cost member variable. |
| getItemNumber | Returns the value in itemNumber. |
| getQuantity | Returns the value in quantity. |
| getCost | Returns the value in cost. |
| getTotalCost | Computes and returns the totalCost. |

**Demonstrate the class by writing a simple program that uses it**. This program should **validate the user inputs** to ensure that negative values are not accepted for item number, quantity, or cost.

Then design a CashRegister class that can be used with the InventoryItem class. The CashRegister class should perform the following:

* 1. Ask the user for the item and quantity being purchased.
  2. Get the item’s cost from the InventoryItem object.
  3. Add a 30% profit to the cost to get the item’s unit price.
  4. Multiply the unit price times the quantity being purchased to get the purchase subtotal.
  5. Compute a 6% sales tax on the subtotal to get the purchase total.
  6. Display the purchase subtotal, tax, and total on the screen.
  7. Subtract the quantity being purchased from the onHand variable of the InventoryItem class object.

**Implement both classes in a complete program**. **Demonstrate the classes by writing a simple program that uses it**. Feel free to modify the InventoryItem class in any way necessary.

**Input Validation:** Do not accept a negative value for the quantity of items being purchased.

1. **(10%)Car Class**

Write a class named Car that has the following member variables:

* year. An int that holds the car’s model year.
* make. A string that holds the make of the car.
* speed. An int that holds the car’s current speed.

In addition, the class should have the following member functions.

* Constructor. The constructor should accept the car’s year and make as arguments and assign these values to the object’s year and make member variables. The constructor should initialize the speed member variable to 0.
* Accessors (“i.e. get function”). Appropriate accessor functions should be created to allow values to be retrieved from an object’s year, make, and speed member variables.
* accelerate. The accelerate function should add 5 to the speed member variable each time it is called.
* brake. The brake function should subtract 5 from the speed member variable each time it is called.

**Demonstrate the class** in a program that creates a Car object, and then calls the accelerate function five times. After each call to the accelerate function, get the current speed of the car and display it. Then, call the brake function five times. After each call to the brake function, get the current speed of the car and display it.

1. **(10%) POSTNET Bar Codes**

The bar code on an envelope used by the US Postal Service represents a five (or more) digit zip code using a format called POSTNET (this format is being deprecated in favor of a new system, OneCode, in 2009). The bar code consists of long and short bars as shown below:



For this program we will represent the bar code as a string of digits. The digit 1 represents a long bar and the digit 0 represents a short bar. Therefore, the bar code above would be represented in our program as:

110100101000101011000010011

The first and last digits of the bar code are always 1. Removing these leave 25 digits. If these 25 digits are split into groups of five digits each then we have:

10100 10100 01010 11000 01001

Next, consider each group of five digits. There will always be **exactly two 1’s in each group of digits**. Each digit stands for a number. From left to right the digits encode the values 7, 4, 2, 1, and 0. Multiply the corresponding value with the digit and compute the sum to get the final encoded digit for the zip code. The table below shows the encoding for 10100.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Bar Code Digits* | *1* | *0* | *1* | *0* | *0* |
| *Value* | *7* | *4* | *2* | *1* | *0* |
| *Product of Digit \* Value* | *7* | *0* | *2* | *0* | *0* |

Zip Code Digit = 7 + 0 + 2 + 0 + 0 = **9**

Repeat this for each group of five digits and concatenate to get the complete zip code. There is one special value. **If the sum of a group of five digits is 11, then this represents the digit 0** (this is necessary because with two digits per group it is not possible to represent zero). The zip code for the sample bar code decodes to 99504. While the POSTNET scheme may seem unnecessarily complex, its design allows machines to detect if errors have been made in scanning the zip code.

Write a zip code class that encodes and decodes five digit bar codes used by the US Postal Service on envelopes. The class should have **two constructors**. The first constructor should input the zip code as an integer and the second constructor should input the zip code as a bar code string consisting of 0’s and 1’s as described above. Although you have two ways to input the zip code, internally the class should only store the zip code using one format (you may choose to store it as a bar code string or as a zip code number.) The class should also have **at least two public member functions**, one to return the zip code as an integer, and the other to return the zip code in bar code format as a string. All helper (member) functions should be declared private. Embed your class definition in a suitable test program.

1. **(10%) Patient Fees**

Write a program that computes a patient’s bill for a hospital stay (including 1. the number of days spent in the hospital and day rate, 2. charges for hospital services (lab tests, surgery charges, etc., and 3. hospital medication charges). The different components of the program are

·The **PatientAccount class** will keep a total of the patient’s charges. It will also keep track of the number of days spent in the hospital. The group must decide on the hospital’s daily rate.

·The **Surgery class** will have stored within it the charges for at least five types of surgery. It can update the charges variable of the PatientAccount class.

·The **Pharmacy clas**s will have stored within it the price of at least five types of medication. It can update the charges variable of the PatientAccount class.

·The main program.

The main program will design a menu that allows the user to enter a type of surgery, enter one or more types of medication, and check the patient out of the hospital. When the patient checks out, the total charges should be displayed.

1. **(10%) Moving inchworm**

Write a program that displays an inchworm on the left-hand side of the screen, facing right. Then slowly move him across the screen, until he disappears off the right-hand side. You may wish to do this in a loop so that after disappearing to the right, the worm appears again on the left. The diagram below shows how he may look at various points on the screen.

