

MÔN HỌC LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

BÀI THỰC HÀNH

CHƯƠNG 1: LỚP VÀ ĐỐI TƯỢNG

Bài 1: Viết lớp Rectangle có các thành viên dữ liệu **private** là **len** và **wid** (lần lượt biểu diễn chiều dài và chiều rộng của hình chữ nhật, là các số không âm). Lớp này có:

- Một constructor nhận 02 đối số nguyên biểu diễn chiều dài và chiều rộng (lớn hơn hoặc bằng 0 và nhỏ hơn hoặc bằng 20) và cả 2 đều có giá trị mặc định là 10.
- Các phương thức public để đặt chiều dài và chiều rộng (các phương thức "set").
- Các phương thức public để lấy chiều dài và chiều rộng (các phương thức "get").
- Các phương thức public để tính và trả về chu vi, đường chéo (số thực double), và diện tích của hình chữ nhật.
- Một phương thức public để xuất hình chữ nhật ra màn hình theo dạng [dài x rộng].
- Một phương thức public để xác định một hình chữ nhật có là hình vuông hay không ?
- Một phương thức public để vẽ hình chữ nhật ra màn hình theo một ký tự chỉ định (hình chữ nhật đặc).

Ghi chú:

- Tập tin rect.h chứa giao tiếp (interface) public cho lớp Rectangle.
- Tập tin rect.cpp chứa cài đặt (implementation) private của các hàm thành viên.
- Tập tin rectTest.cpp là tập tin của client/uer dùng để test lớp Rectangle. Tập tin này có nội dung như sau:

```
#include <iostream>
```

```
#include <cstdlib>
```

```
using namespace std;
```

```
#include "rect.h"
```

```
int main( void )
```

```
{
```

Lập trình hướng đối tượng

```
Rectangle a;          // test rectangle -- default arguments
Rectangle b(12, 20);   // specified length and width
Rectangle c(-5, 15);   // invalid length
```

```
cout << "Rectangle A = ";
a.display();
cout << endl;
```

```
cout << "Rectangle B = ";
b.display();
cout << endl;
```

```
cout << "Rectangle C = ";
c.display();
cout << endl << endl;
```

```
int newLen, newWid;    // input vars for update
```

```
cout << "Enter new length and width for rectangle A: ";
cin >> newLen >> newWid;
a.setLength( newLen );
a.setWidth( newWid );
cout << "New rectangle A: ";
a.display();
cout << endl;
cout << "Perimeter = " << a.perimeter() << endl;
cout << "Diagonal = " << a.diagonal() << endl;
cout << "Area =      " << a.area() << endl;
cout << "Rectangle A is ";
```

```
if( !a.isSquare() )
    cout << "not ";
```

Lập trình hướng đối tượng

```
cout << "a square." << endl << endl;

cout << "Rectangle A:" << endl;
a.draw( '*' );

cout << endl;
return 0;
}
```

Bài 2: Sửa lại định nghĩa và cài đặt của lớp Rectangle của bài 1 sao cho:

Có thể làm việc trên các đối tượng hằng.

Các hàm thành viên không làm thay đổi dữ liệu thành viên private vẫn làm việc được trên các đối tượng hằng.

Thêm một thành viên dữ liệu tĩnh (**private static int**) là **thisMany**. Thành viên này sẽ được tăng lên 1 mỗi khi có một đối tượng hình chữ nhật được tạo và giảm xuống 1 mỗi khi có một đối tượng hình chữ nhật thoát khỏi một phạm vi. Thành viên dữ liệu này được truy cập bởi một hàm thành viên public là **howMany()** cho biết hiện tại số đối tượng của lớp là bao nhiêu?

Thêm một thành viên dữ liệu private là **char *fillString** chứa chuỗi ký tự. Chuỗi này sẽ được truyền bằng tham chiếu khi tạo đối tượng. Giá trị mặc định của chuỗi là chuỗi rỗng "" . Ví dụ, một đối tượng hình chữ nhật được tạo như sau:

```
Rectangle z(15, 10, "Hello");
```

Trong trường hợp này đối tượng hình chữ nhật có chiều dài là 15, chiều rộng là 10 và chuỗi fillString là "Hello".

Thêm một hàm thành viên là **fill** để trả về giá trị của **fillString**.

Không thêm hàm thành viên thay đổi giá trị của **fillString**.

Ghi chú:

- Tập tin rect2.h chứa giao tiếp (interface) public cho lớp Rectangle.
- Tập tin rect2.cpp chứa cài đặt (implementation) private của các hàm thành viên.
- Tập tin rect2Test.cpp là tập tin của client/uer dùng để test lớp Rectangle. Tập tin này có nội dung như sau:

```
#include <iostream>
```

Lập trình hướng đối tượng

```
#include <cstdlib>
using namespace std;

#include "rect2.h"

int main( void )
{
    cout << "Starting # of Rectangles: " << Rectangle::howMany()
        << endl << endl;

    Rectangle a;
    cout << "Rectangle A = ";
    a.display();
    cout << endl << "Now this many Rectangles: " << a.howMany();
    cout << endl << endl;

    Rectangle b(12, 20, "Rectangle B");
    cout << "Rectangle B = ";
    b.display();
    cout << endl << "Now this many Rectangles: " << b.howMany();
    cout << endl << endl;

    { // block to test destructor
        const Rectangle c(-5, 15, "Rectangle C");
        cout << "Rectangle C = ";
        c.display();
        cout << endl << "Now this many Rectangles: " << c.howMany();

        // testing const functions and new fill function
        cout << endl << "Perimeter of Rectangle C is: " << c.perimeter();
        cout << endl << "Fill string of Rectangle C is: " << c.fill();
        cout << endl << endl;
    }
```

Lập trình hướng đối tượng

```
}  
    // should be one less now, since block exited  
    cout << "After exiting block, this many Rectangles: "  
    << Rectangle::howMany() << endl << endl;  
  
    return 0;  
}
```

Bài 3: Xây dựng lớp ngày tháng năm (Date) gồm:

Các thành viên dữ liệu:

Ngày (day) là số nguyên từ 1 đến 31 tùy theo tháng

Tháng (month) là số nguyên từ 1 đến 12

Năm (year) là số nguyên

Các hàm thành viên:

Constructor với 3 tham số là ngày tháng năm với giá trị mặc định là 1/1/1900

Destructor

Hàm xuất ngày tháng năm theo dạng: ngày/tháng/năm

Hàm kiểm tra ngày tháng năm có hợp lệ không ? (Kiểm tra theo ngày, tháng và năm)

Xây dựng lớp nhân viên (Employee) gồm:

Các thành viên dữ liệu:

Họ nhân viên (last name) là chuỗi

Tên nhân viên (first name) là chuỗi

Ngày sinh (birth date) kiểu Date

Ngày vào làm (hire date) kiểu Date

Các hàm thành viên:

Constructor với 4 tham số là họ, tên, ngày sinh, ngày vào làm

Destructor

Hàm xuất đối tượng nhân viên ra màn hình bao gồm: họ, tên, ngày sinh, ngày vào làm

Ghi chú:

- Tập tin date.h chứa giao tiếp (interface) public cho lớp Date.
- Tập tin date.cpp chứa cài đặt (implementation) private của các hàm thành viên.

Lập trình hướng đối tượng

- Tập tin employee.h giao tiếp (interface) public cho lớp Employee.
- Tập tin employee.cpp chứa cài đặt (implementation) private của các hàm thành viên.
- Tập tin employeeTest.cpp của client/uer dùng để test lớp Employee. Tập tin này có nội dung như sau:

// Demonstrating composition--an object with member objects.

```
#include <iostream>
```

```
using std::cout;
```

```
using std::endl;
```

```
#include "Employee.h" // Employee class definition
```

```
int main()
```

```
{
```

```
    Date birth( 7, 24, 1949 );
```

```
    Date hire( 3, 12, 1988 );
```

```
    Employee manager( "Bob", "Blue", birth, hire );
```

```
    cout << endl;
```

```
    manager.print();
```

```
    cout << "\nTest Date constructor with invalid values:\n";
```

```
    Date lastDayOff( 14, 35, 1994 ); // invalid month and day
```

```
    cout << endl;
```

```
    return 0;
```

```
} // end main
```