

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

BÀI TẬP TUẦN 6

2022 – 2023

Bài tập thực hành Inheritance and Polymorphism

Bài 1 Xây dựng một phần mềm vẽ hình (dạng vector). Hiện tại, phần mềm hỗ trợ các loại đối tượng hình vẽ cơ bản sau: hình tròn, hình chữ nhật, tam giác, và đa giác bất kỳ. từ những đối tượng hình vẽ cơ bản, người sử dụng có thể gom nhóm lại thành đối tượng hình vẽ phức tạp.

Gọi Shape là lớp đối tượng hình vẽ tổng quát.

Vẽ Class Diagram của chương trình – word, *publisher*, powerpoint.

```
/* *****  
 * Define Shape class, File giao diện cho lop Shape  
 ***** */  
#ifndef SHAPE_H  
#define SHAPE_H  
#include <iostream>  
using namespace std;  
  
// Định nghĩa diem Point  
struct Point {  
    double X, Y;  
};  
// Định nghĩa lop Shape  
class Shape {  
public:  
    // it'll reimplement in each derived class  
  
    /* area of shape, e.g. area of circle, or retangle */  
    virtual double area() = 0;  
  
    virtual Shape*Clone() = 0;  
    virtual bool IsSelected(Point p) = 0;  
  
    virtual void input() { /*nothing to do*/ };  
    virtual void print() { /*nothing to do*/ };  
};  
#endif  
-----
```



```
/* *****  
 * Define Circle class, File giao dien cho lop Circle  
 * ***** */  
#ifndef CIRCLE_H  
#define CIRCLE_H  
#include "shape.h"  
  
// Dinh nghĩa lớp Circle  
class Circle : public Shape {  
protected:  
    Point C;    // toa do tam  
    double R;    // do dai ban kinh  
public:  
    // Constructors  
    Circle();  
    Circle(Point, double);  
  
    /* override area() method */  
    double area();  
  
    bool IsSelected(Point p);  
    Shape* Clone();  
    void input();  
    void print();  
};  
  
/*----- Implement methods: IsSelected(), Clone() -----*/  
// method IsSelected  
bool Circle::IsSelected(Point p) {  
    // pC <= R : Diem p nam tren hinh tron => IsSelected() = true  
    return ((p.X - C.X)*(p.X - C.X) + (p.Y - C.Y)*(p.Y - C.Y)) <= R*R;  
}  
  
// Deep copy  
Shape* Circle::Clone() {  
    Circle* p = new Circle;  
    if (p == NULL) return NULL;  
    p->C = C;  
    p->R = R;  
    return p;  
}  
  
#endif  
  
-----  
/* *****  
 * Define Rectangle class, File giao dien cho lop Rectangle  
 * ***** */  
#ifndef RECTANGLE_H  
#define RECTANGLE_H  
#include "shape.h"  
  
// Dinh nghĩa lớp Rectangle  
class Rectangle : public Shape {  
protected:  
    Point X;    // Toa do goc trai tren  
    double W, H; // Chieu rong va chieu cao
```



```
public:
    // Constructors
    Rectangle();
    Rectangle(Point, double, double);

    /* area of rectangle */
    double area();

    bool IsSelected(Point p);
    Shape* Clone();
    void input();
    void print();
};
#endif
-----

/*****
 * Define ComplexShape, File giao diện cho lớp ComplexShape
 *****/
#include "shape.h"

// Định nghĩa lớp ComplexShape
class ComplexShape : public Shape {
protected:
    // (1) vector<Shape*> Children;

    // Các hình vẽ thành phần: Circle, rectangle, ...
    int sz; // Number of shapes
    Shape** Children; // Danh sách các hình
public:
    // Constructors

    /* area of ComplexShape */
    double area();

    bool IsSelected(Point p);
    Shape* Clone();

    void input();
    void print();
    int circle_Num();
};
```

🔴(1) Tip: Trên lớp có thể sử dụng <vector> để tạo danh sách các hình 'Children' trong lớp 'ComplexShape'.

Về nhà sử dụng mảng động tự định nghĩa để tạo danh sách các hình 'Children' - như đã khai báo trong lớp 'ComplexShape' - để làm.

Yêu cầu

1. Viết các hàm dựng và hàm hủy và hàm sao chép của các lớp Shape, Circle, Rectangle và ComplexShape.

2. Viết các hàm thành phần khác đã cho trong thiết kế lớp, IsSelected(), và Clone() của các lớp con Circle và Rectangle.
 3. Cài đặt phương thức IsSelected, và Clone cho lớp ComplexShape
 4. Viết hàm nhập/xuất cho lớp ComplexShape để nhập danh sách các hình tròn và hình chữ nhật.
 5. Viết hàm tính số lượng các hình tròn, hình chữ nhật có trong mảng các hình đã nhập.
 6. Viết hàm tính tổng diện tích các hình trong danh sách các hình đã nhập 'Children'.
-

Bài 2 Công ty ABC cần xây dựng ứng dụng quản lý thông tin và tính lương cho nhân viên, sử dụng Kế thừa và Đa hình trong lập trình hướng đối tượng của C++. Thông tin mỗi nhân viên bao gồm: mã nhân viên (C-string), họ tên (C-string), ngày sinh, địa chỉ (C-string).

Công ty có 2 loại nhân viên với cách tính lương như sau:

- Lương(Nhân viên sản xuất): số sản phẩm * 50.000 VNĐ
- Lương(Nhân viên công nhật): số ngày * 500.000 VNĐ

Yêu cầu

- I. Vẽ sơ đồ quan hệ kế thừa giữa các lớp, với mỗi lớp có thông tin chi tiết gồm thành phần dữ liệu và hàm thành phần.

Note: Tương tự như Hình 1.1, Bài 1.

- II. Áp dụng tính kế thừa, khai báo class NVSanXuat và NVCongNhat kế thừa class NhanVien và cài đặt các hàm sau:

1. Cài đặt các Constructors cho mỗi Class
2. Nhập thông tin nhân viên, với ràng buộc dữ liệu nhập như sau:
 - 2.1. Mã nhân viên là chuỗi gồm 5 ký tự số
 - 2.2. Tuổi nhân viên với qui định sau:
 - + Nam: 18 – 60 tuổi
 - + Nữ: 18 – 55 tuổi
 - 2.3. Số sản phẩm: 10-15 sản phẩm
 - 2.4. Số ngày công: 22 – 26 ngày
3. Xuất thông tin nhân viên ra màn hình
4. Tính lương nhân viên

- III. Cài đặt class CongTy sử dụng mảng động hoặc danh sách liên kết. Viết các Constructors và Assignment Operator (toán tử gán bằng), cài đặt các phương thức để thực hiện các chức năng sau.

1. Ghi danh sách nhân viên vào file 'ds_NhanVien.dat'.
2. Đọc danh sách nhân viên từ file 'ds_NhanVien.dat'.



3. Xuất danh sách nhân viên ra màn hình.
 4. Tính tổng tiền lương của tất cả nhân viên.
 5. Tìm nhân viên có lương cao nhất
 6. Tính lương trung bình trong công ty
 7. Nhập vào mã, tìm nhân viên tương ứng
 8. Nhập vào tên, tìm nhân viên tương ứng
 9. Có bao nhiêu nhân viên sinh trong tháng 5
 10. Thêm một nhân viên vào danh sách, và cập nhật lại file 'ds_NhanVien'.
 11. Xóa một nhân viên khỏi danh sách, và cập nhật lại file 'ds_NhanVien'.
 12. Ghi tất cả các nhân viên có lương nhỏ hơn lương trung bình của công ty lên file 'emp_LowerAvgSalary.dat'.
-