

类和对象

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：<https://knights.blog.csdn.net/article/details/107076785>

1. 面向对象的三大概念

- 封装
- 继承
- 多态

2. 类的封装

- 封装是面向对象程序设计最基本的特性
- 概念：把 属性(成员变量) 和 行为(成员函数) 封装为抽象的类，并对属性和行为进行访问控制

3. 类成员的访问控制

- Public 修饰成员变量和成员函数：可以在类的内部和类的外部被访问
- Private修饰成员变量和成员函数：只能在类的内部被访问

4. struct 与 class 关键字区别

- 在用 struct 定义类时，所有成员的默认属性为 public
- 在用 class定义类时，所有成员的默认属性为 private

5. 面向对象程序练习

- 1) 定义一个Point类，其属性包括点的坐标，提供计算两点之间距离的方法；
- 2) 定义一个圆形类，其属性包括圆心和半径；
- 3) 创建两个圆形对象，提示用户输入圆心坐标和半径，判断两个圆是否相交，并输出结果。

```
/* 1 */
class Point{
public:
    void setXY(int x, int y) {
        m_dx = x;
        m_dy = y;
    }
    double getX(){
        return m_dx;
    }
    double getY() {
        return m_dy;
    }

    /* 计算两个坐标之间的距离 */
    /* 公式：|AB| = 根号下 (x1 - x2)^2 + (y1 - y2)^2*/
    double getXtoYLen(Point &m_dp) {
        double dx = m_dp.getX() - m_dx;
        double dy = m_dp.getY() - m_dy;
        return sqrt(dx * dx + dy * dy);
    }

private:
    double m_dx, m_dy;
};
```

```

/* 2 */
class Circle{
public:
    void setCenter(double x, double y) {
        m_dx = x;
        m_dy = y;
    }
    void setRadius(double r) {
        m_dr = r;
    }
    double getX() {
        return m_dx;
    }
    double getY() {
        return m_dy;
    }
    double getR() {
        return m_dr;
    }

    /* 判断两个圆是否相交 0: 不相交 1: 相交 */
    /* 思路: 两个圆心之间的距离 和两个圆的半径之和比较 */
    int detTwoCircleInter(Circle& cir) {
        int ret = 0;
        double dx = cir.getX() - m_dx;
        double dy = cir.getY() - m_dy;
        if ((cir.getR() + m_dr) >= sqrt(dx * dx + dy * dy)) {
            ret = 1; // 两个圆的半径之和 大于等于 两个圆心之间的距离, 说明两个圆相交
        }
        return ret;
    }

private:
    double m_dx, m_dy, m_dr;
};

int main(int argc, const char *argv[]){
    /* 测试 1 */
    Point p1, p2;
    p1.setXY(1, 3);
    p2.setXY(4, 7);
    cout << p1.getXtoYLen(p2) << endl;

    /* 测试 2 */
    Circle c1, c2;
    c1.setCenter(0, 0);
    c1.setRadius(2);
    c2.setCenter(5, 0);
    c2.setRadius(3);

    if (c1.detTwoCircleInter(c2)) {
        cout << "两圆相交" << endl;
    }
    else {
        cout << "两圆不相交" << endl;
    }
    return 0;
}

```

设计并测试一个名为Rectangle的矩形类，其属性为矩形的左下角与右上角两个点的坐标，根据坐标能计算出矩形的面积

```
class Rectangle{
public:
    void setLeftLowXY(int x, int y) {
        m_left_dx = x;
        m_left_dy = y;
    }

    void setRightUpXY(int x, int y) {
        m_right_dx = x;
        m_right_dy = y;
    }

    double getRectangleArea() {
        return (m_right_dx - m_left_dx) * (m_right_dy - m_left_dy);
    }

private:
    double m_left_dx;    // 左下角坐标
    double m_left_dy;
    double m_right_dx;   // 右上角坐标
    double m_right_dy;
};
```

👉 写文不易 且行且珍惜 👈
👉 MrWang 👈