

实验三 面向对象初步

1 实验目的

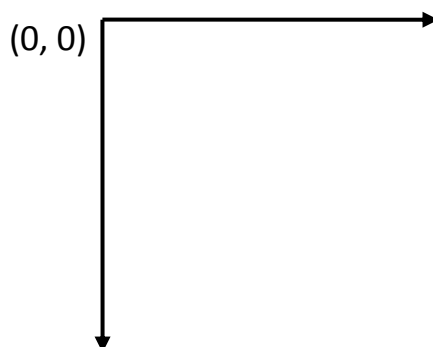
- (1) 学习如何声明和编写类的代码。
- (2) 学习如何编写改变类的属性的成员函数。
- (3) 学习如何声明对象，如何调用对象的成员函数。

2 实验内容

2.1 设计 Point 类（40 分）

(1) 问题描述

计算机的显示屏的坐标系是这样的，左上角的坐标为(0, 0)，如下图所示。



定义计算机显示屏上的点 Point 类。该类具有两个私有数据成员 **x**、**y**，分别表示该点的横坐标、纵坐标。类的声明如下：

```
class Point {  
public:  
    // 默认构造函数，默认值为左上角坐标 (0, 0)  
    Point(int x = 0, int y = 0);  
    void setX(int x);  
    int getX();  
    void setY(int y);  
    int getY();  
    void print();  
    void moveRight(int offset);  
};
```

```

        void moveDown(int offset);
private:
    int x;
    int y;
};
*****

```

(2) 问题要求

请实现以下函数声明，要求能得到如下图所示的运行结果。

- (1) 接受用户的输入，生成两个对象。
- (2) 打印这两个点。
- (3) 向右平移其中一个点后，打印该点。向下平移另一个点后，打印该点。

(3) 主函数代码框架

```

void main(){
    int x, y;
    cout << "Please input a point: ";
    cin >> x >> y;
    Point p1(x,y);          // 生成点对象1
    cout << "Point p1: ";
    p1.print();
    cout << endl;
    Point p2(x * 2, y * 2); //生成点对象2
    cout << "Point p2: ";
    p2.print();
    cout << endl;
    p1.moveRight(10);
    cout << "After moving right, p1: ";
    p1.print();
    cout << endl;
    p2.moveDown(-10);       // 位移量为负数，表示向上移动
    cout << "After moving down, p2: ";
    p2.print();
    cout << endl;
}

```

(4) 运行结果示例

```
Please input a point: 12 8
Point p1: (12, 8)
Point p2: (24, 16)
After moving right, p1: (22, 8)
After moving down, p2: (24, 6)
```

2.2 设计日期类 Date (60 分)

(1) 问题描述

设计一个日期类 **Date**，类的声明如下：

```
class Date{
public:
    /* 默认构造函数，以fullyear的形式给出年月日，默认值为1990年1月1日，同时设置
    日期分隔符为“-” */
    Date(int year = 1990, int month = 1, int day = 1);

    /* get、set方法 */
    // 设置日期，如果有非法的月或日，将其置为1
    void setDate(int year, int month, int day);
    void setYear(int year);
    int getYear();
    void setMonth(int month);
    int getMonth();
    void setDay(int month);
    int getDay();
    void setSeparator(char separator);

    /* 输出函数，请使用setfill('0')和setw(2) */
    void printFullYear(); // 以YYYY-MM-DD的形式打印，2011-01-08
    void printStandardYear(); // 以YY-MM-DD的形式打印，比如11-01-08
    /* 计算函数 */
    // 计算当前日期与参数日期之间相差几个整年，仅考虑参数日期比当前日期晚的情况
    int fullYearsTo(int year, int month, int day);
    /* 计算当前日期与参数日期之间相差多少天(考虑闰年)，如果参数日期在当前日期之前，
```

```

    返回负数。 */
    int daysTo(int year, int month, int day);
private:
    int year;
    int month;
    int day;
    char separator; // 日期分隔符
};

```

(2) 问题要求

请实现日期类，使得主函数及其输出如下：

```

void main(){
    Date birthDate(1969, 8, 11);
    birthDate.printFullYear(); // 打印: 1969-08-11
    birthDate.printStandardYear(); // 打印: 69-08-11
    birthDate.setSeparator('/');
    birthDate.printFullYear(); // 打印: 1969/08/11

    cout << birthDate.fullYearsTo(2010, 4, 15); // 打印: 40, 满四十岁
    cout << birthDate.daysTo(2010, 4, 15); // 打印14857, 活了14857天了
    // 打印-7254, 共和国比我早诞生了7254天
    cout << birthDate.daysTo(1949, 10, 1);
}

```