1、编写几何点（二维平面上）的类Point，包括位置属性（二维坐标x，y），定义带有默认参数的构造函数完成数据成员的初始化，要求x和y的默认值为0；成员函数包括点的位置获取函数GetX（）和GetY（），点的位置设置函数SetX（）和SetY（），和点的位置移动函数MoveTo（）以及点的信息打印函数Display（）。

2、编写几何图形圆的类Circle，包括两个属性：圆心O（用上题中的Point类实现）和半径R。成员函数包括：构造函数、圆心位置获取函数GetO（）和半径获取函数GetR（）；圆心位置设置函数SetO（）；圆的位置移动函数MoveTo（）；圆的半径设置函数SetR（）；以及圆的信息打印函数Display（）。

3、在main函数中定义圆对象，实现半径、圆心的设置、移动和输出操作。

#include <iostream.h>

class Point

{

private:

float X,Y;

public:

Point(float x=0,float y=0)

{

X=x;Y=y;

}

Point(Point &p)

{

X=p.X;

Y=p.Y;

}

void SetX(float x) {X=x;}

void SetY(float y) {Y=y;}

float GetX() {return X;}

float GetY(){return Y;}

void MoveTo(float x,float y)

{

X= x;Y=y;

}

void Dispaly()

{

cout<<"("<<X<<","<<Y<<")";

}

};

class Circle

{

private:

Point O; //圆心

float R; //半径

public:

Circle(Point p,float r):O(p)

{

R=r;

}

void SetO(Point p) {O=p;}

void SetR(float r) {R=r;}

Point GetO()

{

return O;

}

float GetR()

{

return R;

}

void MoveTo(float x,float y)

{

O.SetX(x);

O.SetY(y);

}

void Display()

{

cout<<"圆心在("<<O.GetX()<<","<<O.GetY()<<")半径为"<<R<<endl;

}

};

void main( )

{

float x,y,r;

cout<<"请输入圆的半径：";

cin>>r;

cout<<"请输入圆心的横坐标和纵坐标：";

cin>>x>>y;

Point o(x,y);

Circle c1(o,r);

c1.Display();

cout<<"请输入圆心移动后的横坐标和纵坐标：";

cin>>x>>y;

c1.MoveTo(x,y);

cout<<"移动后，";

c1.Display();

}