实验作业 2

阅读下列程序 1、程序 2、程序 3，写出输出结果，并上机调试这三个程序，分析说明它们输出不同的原因，并总结出输出特点和规律。

程序 1：

```
#include "stdafx.h"
#include <iostream>
#include <string.h>
using namespace std;

class Graphics
{
public:
   virtual double area() = 0;
   virtual void draw() { cout << "Graphics的draw()被调用" << endl; };
   virtual ~Graphics(){ cout << "Graphics类对象被析构" << endl << endl; }
};

class Points : public Graphics
{
protected:
   int x;
   int y;
public:
   Points() :x(0), y(0){};
   Points(int x0, int y0) :x(x0), y(y0){};
   double area(){
      return 0;
   };
   void draw(){
      cout << "Points的draw()被调用" << endl;
      cout << "x=" << x << ",y=" << y << endl;
   }
   ~Points(){ cout << "Points类对象被析构" << endl; }
};

class Circle : public Points
{
private:
   int r;
public:
```

```
    Circle() :Points(), r(0){};
    Circle(int x0, int y0, int r0) :Points(x0, y0), r(r0){};
    double area(){
        return 3.14*r*r;
    };
    void draw()
    {
        cout << "Circle的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y <<",r="<<r<< endl;
        cout << "Circle的面积为: " << area() << endl;
    }
     ~Circle(){ cout << "Circle类对象被析构" << endl; }
};

class Ellipse : public Points
{

private:
    int a;
    int b;
public:
    Ellipse() :Points(), a(0),b(0){};
    Ellipse(int x0, int y0, int a0,int b0) :Points(x0, y0), a(a0),b(b0){};
    double area(){ return 3.14*a*b; };
    void draw()
    {
        cout << "Ellipse的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y << ",a=" << a<<",b=" << b << endl;
        cout << "Ellipse的面积为: " << area() << endl;
    }
    ~Ellipse(){ cout << "Ellipse类对象被析构" << endl; }
};


int main()
{
    Graphics *ptr;
    cout << "Graphics类指针指向派生类: " << endl<<endl;
    ptr = new Points(1, 1);
    ptr->draw();
    delete ptr;

    ptr = new Circle(2, 2, 4);
    ptr->draw();
```

```
    delete ptr;

    ptr = new Ellipse(3, 3, 2, 1);
    ptr->draw();
    delete ptr;

    Points *p_ptr;
    cout << "Points类指针指向派生类: " << endl << endl;
    p_ptr = new Circle(4, 4, 8);
    p_ptr->draw();
    delete p_ptr;

    p_ptr = new Ellipse (6, 6, 4, 2);
    p_ptr->draw();
    delete p_ptr;

    return 0;
}
```

程序 2：

```
#include "stdafx.h"
#include <iostream>
#include <string.h>
using namespace std;

class Graphics
{
public:
    virtual double area() = 0;
    void draw() { cout << "Graphics的draw()被调用" << endl; };
    ~Graphics(){ cout << "Graphics类对象被析构" << endl << endl; }
};

class Points : public Graphics
{
protected:
    int x;
    int y;
public:
    Points() :x(0), y(0){};
    Points(int x0, int y0) :x(x0), y(y0){};
    double area(){
```

```
        return 0;
    };
    virtual void draw(){
        cout << "Points的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y << endl;
    }
    virtual ~Points(){ cout << "Points类对象被析构" << endl; }
};

class Circle : public Points
{
private:
    int r;
public:
    Circle() :Points(), r(0){};
    Circle(int x0, int y0, int r0) :Points(x0, y0), r(r0){};
    double area(){
        return 3.14*r*r;
    };
    void draw()
    {
        cout << "Circle的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y <<",r="<<r<< endl;
        cout << "Circle的面积为: " << area() << endl;
    }
     ~Circle(){ cout << "Circle类对象被析构" << endl; }
};

class Ellipse : public Points
{

private:
    int a;
    int b;
public:
    Ellipse() :Points(), a(0),b(0){};
    Ellipse(int x0, int y0, int a0,int b0) :Points(x0, y0), a(a0),b(b0){};
    double area(){ return 3.14*a*b; };
    void draw()
    {
        cout << "Ellipse的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y << ",a=" << a<<",b=" << b << endl;
        cout << "Ellipse的面积为: " << area() << endl;
    }
```

```
    ~Ellipse(){ cout << "Ellipse类对象被析构" << endl; }
};


int main()
{
    Graphics *ptr;
    cout << "Graphics类指针指向派生类： " << endl<<endl;
    ptr = new Points(1, 1);
    ptr->draw();
    delete ptr;

    ptr = new Circle(2, 2, 4);
    ptr->draw();
    delete ptr;

    ptr = new Ellipse(3, 3, 2, 1);
    ptr->draw();
    delete ptr;

    Points *p_ptr;
    cout << "Points类指针指向派生类： " << endl << endl;
    p_ptr = new Circle(4, 4, 8);
    p_ptr->draw();
    delete p_ptr;

    p_ptr = new Ellipse (6, 6, 4, 2);
    p_ptr->draw();
    delete p_ptr;

    return 0;
}
```

程序 3：

```
#include "stdafx.h"
#include <iostream>
#include <string.h>
using namespace std;

class Graphics
{
public:
    double area() = 0;
    void draw() { cout << "Graphics的draw()被调用" << endl; };
```

```
    ~Graphics(){ cout << "Graphics类对象被析构" << endl << endl; }
};

class Points : public Graphics
{
protected:
    int x;
    int y;
public:
    Points() :x(0), y(0){};
    Points(int x0, int y0) :x(x0), y(y0){};
    double area(){
        return 0;
    };
    void draw(){
        cout << "Points的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y << endl;
    }
    ~Points(){ cout << "Points类对象被析构" << endl; }
};

class Circle : public Points
{
private:
    int r;
public:
    Circle() :Points(), r(0){};
    Circle(int x0, int y0, int r0) :Points(x0, y0), r(r0){};
    double area(){
        return 3.14*r*r;
    };
    void draw()
    {
        cout << "Circle的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y <<",r="<<r<< endl;
        cout << "Circle的面积为: " << area() << endl;
    }
     ~Circle(){ cout << "Circle类对象被析构" << endl; }
};

class Ellipse : public Points
{

private:
```

```
    int a;
    int b;
public:
    Ellipse() :Points(), a(0),b(0){};
    Ellipse(int x0, int y0, int a0,int b0) :Points(x0, y0), a(a0),b(b0){};
    double area(){ return 3.14*a*b; };
    void draw()
    {
        cout << "Ellipse的draw()被调用" << endl;
        cout << "x=" << x << ",y=" << y << ",a=" << a<<",b=" << b << endl;
        cout << "Ellipse的面积为: " << area() << endl;
    }
    ~Ellipse(){ cout << "Ellipse类对象被析构" << endl; }
};


int main()
{
    Graphics *ptr;
    cout << "Graphics类指针指向派生类: " << endl<<endl;
    ptr = new Points(1, 1);
    ptr->draw();
    delete ptr;

    ptr = new Circle(2, 2, 4);
    ptr->draw();
    delete ptr;

    ptr = new Ellipse(3, 3, 2, 1);
    ptr->draw();
    delete ptr;

    Points *p_ptr;
    cout << "Points类指针指向派生类: " << endl << endl;
    p_ptr = new Circle(4, 4, 8);
    p_ptr->draw();
    delete p_ptr;

    p_ptr = new Ellipse (6, 6, 4, 2);
    p_ptr->draw();
    delete p_ptr;

    return 0;
}
```