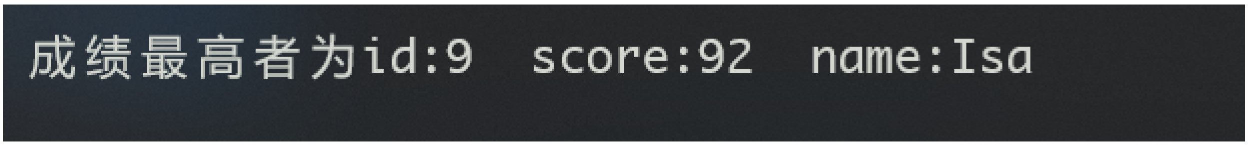
第三次作业

第一题

```cpp
#include <iostream>
using namespace std;
class Student {
public:
    Student(int id,float score,string name) :id(id),score(score),name(name) {}
    ~Student() {}
    Student(const Student& rhs) :id(rhs.id),score(rhs.score),name(rhs.name) {}
    int id;
    float score;
    string name;
private:
};
int max(Student* p);
int main() {
    Student s[10] = { Student(1,80,"Alice"),Student(2,90,"Bob"), Student(3,70,"Carol"), Student(4,60,"Dave"), Student(5,75,"Eve"),Student(6,80,"Francis"),Student(7,65,"David"), Student(8,77,"Justin"), Student(9,92,"Isa"), Student(10,88,"Lindy")};
    Student* p = &s[0];
    int max_id = max(p);
    Student copy(s[max_id - 1]);
    cout << "成绩最高者为id:" << max_id << "  " << "score:" << copy.score << "  " << "name:" << copy.name<<endl;
}

int max(Student* p) {
    int tscore = p->score;
    int tid = p->id;
    for (int i = 0; i < 10; i++) {
        if (p->score >= tscore) {
            tscore = p->score;
            tid = p->id;
        }
        p++;
    }
    return tid;
}

```



```
成绩最高者为id:9  score:92  name:Isa
```

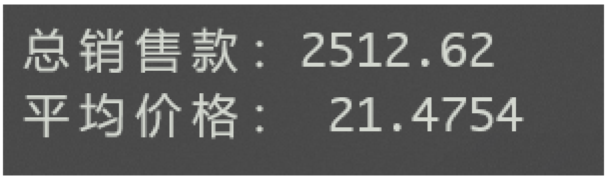
第二题

```
#include <iostream>
using namespace std;
class Shop {
public:
    static int id[3];
    static int piece_num[3];
    static float price[3];
    Shop()  {
    }

    ~Shop() {
    }
    static float sum() {
        return piece_num[0] * price[0] + piece_num[1] * price[1] * 0.98 + piece_num[2] * price[2] * 0.98;
    }
    static float avg_price() { return sum() / (piece_num[0] + piece_num[1] + piece_num[2]); }
```

```cpp
private:

};
int Shop::id[3] = { 101,102,103 };
int Shop::piece_num[3] = { 5,12,100 };
float Shop::price[3] = { 23.5,24.5,21.5 };
int main() {
    cout << "总销售款：" << Shop::sum() << endl << "平均价格：  " << Shop::avg_price() << endl;
}
```



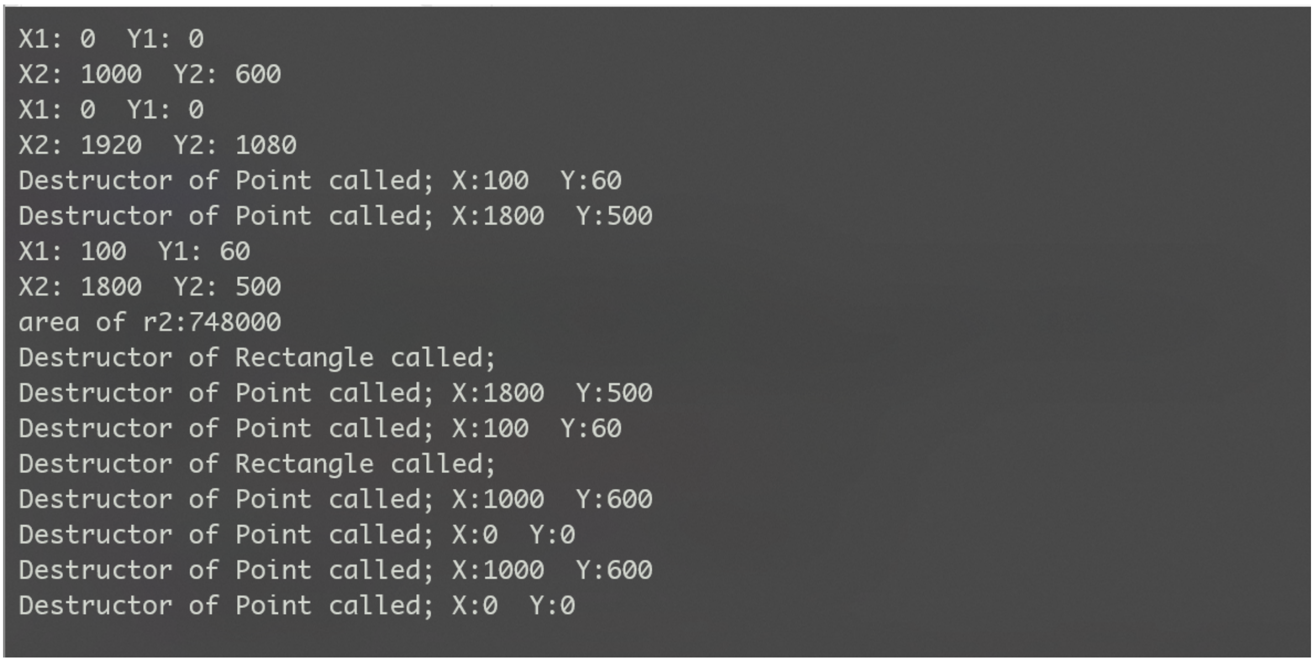
```
总销售款：2512.62
平均价格：  21.4754
```

第三题

```cpp
#include <iostream>
#define MAX_X 1920
#define MAX_Y 1080
using namespace std;
class Point {
public:
    Point(int x=0,int y=0) :_x(x),_y(y) {
    }
    Point(const Point& rhs):_x(rhs._x),_y(rhs._y) {}
    ~Point() {
        cout << "Destructor of Point called; X:" << _x << "  " << "Y:" << _y<<endl;
    }
    inline void set(int x, int y) {
        _x = x;
        _y = y;
    }
    inline int get_x() const{ return _x; }
    inline int get_y() const{ return _y; }
    void print() const { cout << "X: " << _x << "  " << "Y: " << _y<<endl; }

private:
    int _x, _y;
};
class Rectangle {
public:
    Rectangle(Point &p1,Point &p2):min(p1),max(p2) {}
    Rectangle(int x1=0,int y1=0,int x2=MAX_X,int y2=MAX_Y):min(x1,y1),max(x2,y2) {}
    Rectangle(const Rectangle& rhs):min(rhs.min),max(rhs.max) {}
    ~Rectangle() {
        cout << "Destructor of Rectangle called;" << endl;
    }
    inline void set(const Point& min,const  Point& max) {
        this->min = min;
        this->max = max;
    }
    inline void print() const { cout << "X1: " << min.get_x() << "  " << "Y1: " << min.get_y() << endl << "X2: " << max.get_x() << "  " << "Y2: " << max.get_y() << endl; }
    inline int area()const {
        return (max.get_x() - min.get_x()) * (max.get_y() - min.get_y());
    }
private:
    Point min, max;
};

int main() {
    Point min(0, 0),max(1000,600);
    Rectangle r1(min, max);
    Rectangle r2;
    r1.print();
    r2.print();
    r2.set(Point(100, 60), Point(1800, 500));
    r2.print();
    cout << "area of r2:" << r2.area() << endl;
}
```



```
X1: 0  Y1: 0
X2: 1000  Y2: 600
X1: 0  Y1: 0
X2: 1920  Y2: 1080
Destructor of Point called; X:100  Y:60
Destructor of Point called; X:1800  Y:500
X1: 100  Y1: 60
X2: 1800  Y2: 500
area of r2:748000
Destructor of Rectangle called;
Destructor of Point called; X:1800  Y:500
Destructor of Point called; X:100  Y:60
Destructor of Rectangle called;
Destructor of Point called; X:1000  Y:600
Destructor of Point called; X:0  Y:0
Destructor of Point called; X:1000  Y:600
Destructor of Point called; X:0  Y:0
```