1. **填空题**(每空1分，共14分)

（1）p1.show(); p->show() （2）公有、私有、保护 （3）class （4）抽象类

（5）成员函数、不是 （6）friend （7）11、11 （8）动态申请内存空间、释放由new申请的空间

1. **选择题**(每小题1.5分，共30分)

1、D 2、B 3、C 4、D 5、D 6、B 7、B 8、C 9、C 10、A 11、D 12、C 13、D 14、B 15、C 16、A 17、B 18、C 19、C 20、A

1. **改错题**(每错2分，共6分)

MyClass obj1(0); obj1.SetMember(5); obj2.SetMember(10);

1. **写出下列程序的执行结果**（每小题5分，共20分）

（1）30 （5分）

（2）7 9 (每个2.5分)

（3）20 20 (每个2.5分)

（4）

3（2分）

3（2分）

3（1分）

1. **编程题**（每题10分、共30分）

1、

#include <iostream.h>

#include <math.h>

class rectangle //(2分)

{

private:

int x1,y1,x2,y2; // (2分)

public:

rectangle(int xx1,int yy1,int xx2,int yy2) //(1分)

{

x1=xx1;y1=yy1;x2=xx2;y2=yy2;

}

int getarea() //(2分)

{

return abs((x2-x1)\*(y1-y2));

}

};

void main()

{

rectangle rect1(3,7,8,5); (2分)

cout<<rect1.getarea()<<endl; (1分)

}

2、(10分)

#include <iostream.h>

class car;（1分）

class boat{

private:

int weight; //(1分)

public:

boat(int w):weight(w){} //(1分)

friend int totalweight(boat b1,car c1); //(2分)

};

class car{ /(1分)

private:

int weight; (1分)

public:

car(int w):weight(w){};

friend int totalweight(boat b1,car c1); (1分)

};

int totalweight(boat b1,car c1) //(1分)

{

return b1.weight+c1.weight;

}

void main()

{

car c1(1000);

boat b1(2000);

cout<<totalweight(b1,c1)<<endl;(1分)

}

3、(10分)

#include<iostream.h>

class vehicle // 定义汽车类 (3分)

{

protected:

int wheels; // 车轮数

float weight; // 重量

public:

vehicle(int wheels,float weight);

int get\_wheels();

float get\_weight();

float wheel\_load();

void show();

};

class car:public vehicle // 定义小车类 (3分)

{

int passenger\_load; // 载人数

public:

car(int wheels,float weight,int passengers=4);

int get\_passengers();

void show();

};

vehicle::vehicle(int wheels1,float weight1) //(1分)

{

wheels=wheels1;

weight=weight1;

}

int vehicle::get\_wheels()

{

return wheels;

}

float vehicle::get\_weight()

{

return weight;

}

void vehicle::show() (1分)

{

cout << "车轮:" << wheels << "个" << endl;

cout << "重量:" << weight << "公斤" << endl;

}

car::car(int wheels, float weight,

int passengers) :vehicle(wheels, weight)

{

passenger\_load=passengers;

}

int car::get\_passengers ()

{

return passenger\_load;

}

void car::show()

{

cout <<" 车型:小车" << endl;

vehicle::show();

cout << "载人:" << passenger\_load << "人" << endl;

cout << endl;

}

void main ()

{

car car1(4,2000,5); (1分)

cout << "输出结果" << endl;

car1. show (); (1分)

}