Условие:

Создать абстрактный класс CVehicle. На его основе реализовать классы CPlane и ССаr. Классы должны иметь возможность задавать и получать координаты, параметры средств передвижения (цена, скорость, год выпуска). Для самолета должна быть определена высота, для автомобиля — количество пассажиров. Написать программу, создающую список объектов этих классов в динамической памяти.

Исходный код программы:#include **<iostream>**#include **<list>  
using namespace** std;  
  
**class** CVehicle{  
**protected**:  
 **int** cost;  
 **int** speed;  
 **int** year;  
};  
  
**class** CCar : **public** CVehicle{  
**public**:  
 **int** passagers;  
  
 **void** SetParams(){  
 cout << **"Cost="**;  
 cin >> cost;  
 cout << **"Speed="**;  
 cin >> speed;  
 cout << **"Year="**;  
 cin >> year;  
 }  
  
 **void** GetParams(){  
 cout << **"Cost="** << cost << **"\n"**;  
 cout << **"Speed="** << speed << **"\n"**;  
 cout << **"Year="** << year << **"\n"**;  
 }  
};  
  
**class** CPlane : **public** CVehicle{  
**public**:  
 **int** height;  
  
 **void** SetParams(){  
 cout << **"Cost="**;  
 cin >> cost;  
 cout << **"Speed="**;  
 cin >> speed;  
 cout << **"Height="**;  
 cin >> height;  
 }  
  
 **void** GetParams(){  
 cout << **"Cost="** << cost << **"\n"**;  
 cout << **"Speed="** << speed << **"\n"**;  
 cout << **"Height="** << height << **"\n"**;  
 }  
};  
  
**int** main()  
{  
 **const int** n = 2;  
  
 CCar \*cars[n];  
 CPlane \*planes[n];  
  
 **for** (**int** i = 0; i < n; i++){  
 cout << **"CCar number "** << i << **"\n"**;  
 cars[i] = **new** CCar();  
 cars[i]->SetParams();  
 }  
  
 **for** (**int** i = 0; i < n; i++){  
 cout << **"CPlane number "** << i << **"\n"**;  
 planes[i] = **new** CPlane();  
 planes[i]->SetParams();  
 }  
  
 **for** (**int** i = 0; i < n; i++){  
 cout << **"CCar number "** << i << **"\n"**;  
 cars[i]->GetParams();  
 }  
  
 **for** (**int** i = 0; i < n; i++){  
 cout << **"CPlane number "** << i << **"\n"**;  
 planes[i]->GetParams();  
 }  
  
 **return** 0;  
}

Скриншоты программы:

