

9^η Φροντιστηριακή Άσκηση στη C++: Κληρονομικότητα

Άσκηση 1: Βασική Κληρονομικότητα

Εκφώνηση:

Γράψτε μια βασική κλάση `Person` με όνομα και ηλικία. Δημιουργήστε μια παράγωγη κλάση `Student` που προσθέτει έναν βαθμό. Υλοποιήστε μεθόδους για την εμφάνιση των δεδομένων.

Λύση:

```
#include <iostream>
using namespace std;

class Person {
public:
    string name;
    int age;
    Person(string n, int a) : name(n), age(a) {}
    void display() {
        cout << "Όνομα: " << name << ", Ηλικία: " << age << endl;
    }
};

class Student : public Person {
public:
    float grade;
    Student(string n, int a, float g) : Person(n, a), grade(g) {}
    void display() {
        Person::display();
        cout << "Βαθμός: " << grade << endl;
    }
};

int main() {
    Student s("Γιάννης", 20, 8.5);
    s.display();
    return 0;
}
```

Άσκηση 2: Πολυεπίπεδη Κληρονομικότητα

Εκφώνηση:

Γράψτε μια βασική κλάση `Animal` που περιέχει ένα όνομα. Δημιουργήστε μια παράγωγη κλάση `Mammal` που προσθέτει μια μέθοδο για τον ήχο και μια τρίτη κλάση `Dog` που προσθέτει μια μέθοδο εμφάνισης πληροφοριών.

Λύση:

```
#include <iostream>
using namespace std;

class Animal {
public:
    string name;
    Animal(string n) : name(n) {}
};

class Mammal : public Animal {
public:
    Mammal(string n) : Animal(n) {}
    void sound() {
        cout << name << " κάνει ήχο!" << endl;
    }
};

class Dog : public Mammal {
public:
    Dog(string n) : Mammal(n) {}
    void display() {
        cout << "Σκύλος: " << name << endl;
        sound();
    }
};

int main() {
    Dog d("Ρεξ");
    d.display();
    return 0;
}
```

Άσκηση 3: Κληρονομικότητα και Κατασκευαστές

Εκφώνηση:

Δημιουργήστε μια βασική κλάση `Vehicle` που περιέχει τον αριθμό τροχών.
Δημιουργήστε μια παράγωγη κλάση `Car` που έχει και μάρκα αυτοκινήτου και χρησιμοποιεί τον constructor της βασικής κλάσης.

Λύση:

```
#include <iostream>
using namespace std;

class Vehicle {
public:
    int wheels;
    Vehicle(int w) : wheels(w) {}
};

class Car : public Vehicle {
public:
    string brand;
    Car(string b, int w) : Vehicle(w), brand(b) {}
    void display() {
        cout << "Μάρκα: " << brand << ", Τροχοί: " << wheels << endl;
    }
};

int main() {
    Car c("Toyota", 4);
    c.display();
    return 0;
}
```

Άσκηση 4: Πολλαπλή Κληρονομικότητα

Εκφώνηση:

Γράψτε δύο κλάσεις `Engine` και `Wheels` που περιέχουν πληροφορίες για τον κινητήρα και τους τροχούς ενός οχήματος. Δημιουργήστε μια κλάση `Car` που κληρονομεί και από τις δύο.

Λύση:

```
#include <iostream>
using namespace std;

class Engine {
public:
    int horsepower;
    Engine(int hp) : horsepower (hp) {}
};

class Wheels {
public:
    int numWheels;
    Wheels(int nw) : numWheels (nw) {}
};

class Car : public Engine, public Wheels {
public:
    string brand;
    Car(string b, int hp, int nw) : Engine (hp), Wheels (nw), brand (b)
    {}
    void display() {
        cout << "Μάρκα: " << brand << ", Ιπποδύναμη: " << horsepower
<< ", Τροχοί: " << numWheels << endl;
    }
};

int main() {
    Car c("BMW", 300, 4);
    c.display();
    return 0;
}
```