

Polymorphism_Example2

May 12, 2020

```
In [ ]: #include <assert.h>
        #include <cmath>
        // TODO: Define PI

        // TODO: Declare abstract class VehicleModel
        class VehicleModel{
        public:
            // TODO: Declare virtual function Move()
            virtual void Move(double v, double phi) = 0;
        };

        // TODO: Derive class ParticleModel from VehicleModel
        class ParticleModel : public VehicleModel{
        public:
            // TODO: Override the Move() function
            void Move(double v, double phi) override{
                theta += phi;
                x += v * cos(theta);
                y += v * sin(theta);
            }
            // TODO: Define x, y, and theta
            double x;
            double y;
            double theta;
        };

        // TODO: Derive class BicycleModel from ParticleModel
        class BicycleModel : public ParticleModel{
        public:
            // TODO: Override the Move() function
            void Move(double v, double phi) override{
                theta += v / L * tan(phi);
                x += v * cos(theta);
                y += v * sin(theta);
            }
            // TODO: Define L
            double L;
        };
```

```
// TODO: Pass the tests
int main() {
    // Test function overriding
    ParticleModel particle;
    BicycleModel bicycle;
    particle.Move(10, M_PI / 9);
    bicycle.Move(10, M_PI / 9);
    assert(particle.x != bicycle.x);
    assert(particle.y != bicycle.y);
    assert(particle.theta != bicycle.theta);
}
```

Compile & Execute

Explain

Loading terminal (id_h177dz1), please wait...