#include <iostream>

#include <cmath>

using namespace std;

class Car {

private:

string make;

string model;

int year;

double speed;

double x;

double y;

double z;

public:

Car(string make, string model, int year, double speed, double x, double y, double z) {

this->make = make;

this->model = model;

this->year = year;

this->speed = speed;

this->x = x;

this->y = y;

this->z = z;

}

void accelerate(double speed\_increment) {

speed += speed\_increment;

}

void brake(double speed\_decrement) {

speed -= speed\_decrement;

if (speed < 0) {

speed = 0;

}

}

void move() {

double distance\_traveled = speed / 3600; // Convert mph to miles per second

x += distance\_traveled;

y += distance\_traveled;

z += distance\_traveled;

}

bool detect\_collision(Car car2) {

double distance\_between\_cars = sqrt(pow((car2.x - x), 2) + pow((car2.y - y), 2) + pow((car2.z - z), 2));

if (distance\_between\_cars < 10) { // 10 feet collision distance

return true;

} else {

return false;

}

}

double time\_to\_collision(Car car2) {

double relative\_speed = speed - car2.speed;

double distance\_between\_cars = sqrt(pow((car2.x - x), 2) + pow((car2.y - y), 2) + pow((car2.z - z), 2));

double time\_to\_collision = distance\_between\_cars / relative\_speed;

return time\_to\_collision;

}

};

int main() {

// Test the Car class

Car car1("Hyundai", "Creta", 2022, 50, 0, 0, 0);

Car car2("BMW", "X5", 2021, 60, 10, 10, 0);

car1.accelerate(20);

cout << "Car1 speed: " << car1.speed << " mph" << endl;

car2.brake(10);

cout << "Car2 speed: " << car2.speed << " mph" << endl;

car1.move();

cout << "Car1 position: (" << car1.x << ", " << car1.y << ", " << car1.z << ")" << endl;

bool collided = car1.detect\_collision(car2);

cout << "Collision detected? " << (collided ? "Yes" : "No") << endl;

double time\_to\_collison = car1.time\_to\_collision(car2);

cout << "Time to collision: " << time\_to\_collison << " seconds" << endl;

return 0;

}