**1050 Programming Logic**Object-Oriented Programming

NAME: **Angela Dennis**

1. Create a project called OOP.
2. Add a class called Car (Leave Program.cs as-is).
3. Add the following properties:

private int Speed;

private string Make;

private string Model;

private int Year;

1. Add two constructor methods:
   1. One that accepts 4 parameters: make, model, year, and speed (in that order).
   2. One that accepts make, model and year (assign a value of 0 to Speed).

Here’s an example of a constructor method that accepts two parameters:

public Car(string make, string model)

{

this.Make = make;

this.Model = model;

}

1. Add methods called SpeedUp() and SlowDown(). These should increment or decrement the value of this.Speed repectively. Only decrement if the value is greater than 0. Both should return the value of Speed after incrementing or decrementing the value.
2. Add a method called Display(). It should not return a value. Display() should output the following:

Console.WriteLine(Year + " " + Make + " " + Model + " is going " + Speed + " MPH.");

1. In Program.cs, copy/paste the following code into the Main method to test your class:

int car1Speed = 20;

int car2Speed = 0;

Car car1 = new Car("Ford", "Focus", 2010, car1Speed);

Car car2 = new Car("Chevy", "Cruze", 2018, car2Speed);

for (int i = 0; i < 60; i++)

{

if (i % 2 == 0)

{

car2Speed = car2.SpeedUp();

}

if (i % 3 == 0)

{

car1Speed = car1.SpeedUp();

}

if (i % 5 == 0)

{

car1Speed = car1.SlowDown();

car2Speed = car2.SlowDown();

}

}

car1.Display();

car2.Display();

**Expected Output:**

2010 Ford Focus is going 28 MPH.

2018 Chevy Cruze is going 18 MPH.

***Code:***

namespace OOP

{

class Program

{

static void Main(string[] args)

{

int car1Speed = 20;

int car2Speed = 0;

Car car1 = new Car("Ford", "Focus", 2010, car1Speed);

Car car2 = new Car("Chevy", "Cruze", 2018, car2Speed);

for (int i = 0; i < 60; i++)

{

if (i % 2 == 0)

{

car2Speed = car2.SpeedUp();

}

if (i % 3 == 0)

{

car1Speed = car1.SpeedUp();

}

if (i % 5 == 0)

{

car1Speed = car1.SlowDown();

car2Speed = car2.SlowDown();

}

}

car1.Display();

car2.Display();

Console.Write("\nPress any key to exit...");

Console.ReadKey(true);

}

}

class Car

{

private int Speed;

private string Make;

private string Model;

private int Year;

public Car(string make, string model, int year, int speed)

{

this.Make = make;

this.Model = model;

this.Year = year;

this.Speed = speed;

}

public Car(string make, string model, int year)

{

this.Make = make;

this.Model = model;

this.Year = year;

this.Speed = 0;

}

public int SpeedUp()

{

Speed++;

return Speed;

}

public int SlowDown()

{

if (Speed > 0)

{

Speed--;

}

return Speed;

}

public void Display()

{

Console.WriteLine(Year + " " + Make + " " + Model + " is going " + Speed + " MPH.");

}

}

}

