```
//Steer.h : contains declaration of Steer class

#ifndef STEER_H
#define STEER_H

class Steer
{
public:

//	Steer();
	Steer(double w, double x, double y, double a);

	void Calc4WheelTurn(double radian);

	void SetA(const double A);

	const double GetFRRatio();
	const double GetFLRatio();
	const double GetRRRatio();
	const double GetRLRatio();

	const double GetThetaFL();
	const double GetThetaFR();
	const double GetThetaRC();
	const double GetThetaRL();
```

```
        const double GetThetaRR();


private:


        void LeftTurn4Wheels();

        void RightTurn4Wheels();


        //Varibles used in class


        //Varibles needed to be defined in constructor

        double pi;      //3.14159

        double W;       //W is the distance between the center of the two back wheels (rear track)

        double X;       //X is the distance between the center of the font and back wheels (wheel base)

        double Y;       //Y is the distance between the center of the two front wheels (front track)

        double A;       //A ratio of X defining the location on the robot to turn about.  (0-1)

                                //e.g. If A=1, robot will turn around the cetner point between the front wheels and front wheel angles do not change.  If A=.5, robot will turn aoround the center point of the wheelbase


        //Varibles calculated and used by class

        double FL;      //FL Turning Radius, distance from Front Left Wheel to the center of rotation (0 to infiity)

        double FR;      //FR Turning Radius, distance from Front Right Wheel to the center of rotation (0 to infiity)

        double RL;      //RL Turning Radius, distance from Rear Left Wheel to the center of rotation (0 to infiity)

        double RR;      //RR Turning Radius, distance from Rear Right Wheel to the center of rotation (0 to infiity)
        double Z;          //Robot Turning Raduis, distance form A * X to center of rotation (0 to infiity)


        //Varibles accessable by user
        double thetaRC;            //Rear Center Wheel Angle, angle of rear center wheel (-pi/2 to 3*pi/2)


        double thetaFL; //Front Left Angle, angle of rear center wheel (-pi/2 to 3*pi/2)
        double thetaFR;//Front Right Angle, angle of rear center wheel (-pi/2 to 3*pi/2)
        double thetaRL;//Rear Left Angle, angle of rear center wheel (-pi/2 to 3*pi/2)
        double thetaRR;            //Rear Right Angle, angle of rear center wheel (-pi/2 to 3*pi/2)


        double FRRatio;//Ratio of Speed of Front Right wheel to Maximum of all wheel speeds (-1 to 1)
        double FLRatio; //Ratio of Speed of Front Left wheel to Maximum of all wheel speeds (-1 to 1)
        double RRRatio;//Ratio of Speed of Rear Right wheel to Maximum of all wheel speeds (-1 to 1)
        double RLRatio; //Ratio of Speed of Rear Left wheel to Maximum of all wheel speeds (-1 to 1)


};


#endif   //STEER_H
```