

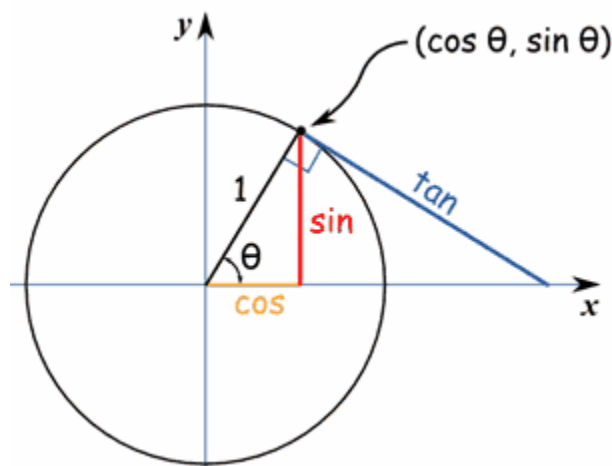
Trigonometry in Car Driving Games with Flash Actionscript

We use Trigonometry math in Flash to be able to make an object (such as a car), go forward and also turn at the same time.

In Flash the movement happens by continually and rapidly changing the x and y coordinates of the car. This is usually done in response to a User pressing certain buttons on the keyboard, such as the arrow keys. Working out the latest X and Y coordinates is easy if we are simply moving in a left to right straight line motion across the screen, because all we have to do is keep adding a fixed value onto the X coordinate. A similar situation exists if we move up the screen in the Y direction.

However, what if we rotate our car so that it starts turning around a corner ? How do we calculate the change in x and y values to get it from position A to position B ?

The answer is that we have to use Sine and Cosine trigonometry. If we rotate and move a distance of 1 unit at a given angle θ , relative to our starting point, then the change in X coordinate value is $1 \times \cos\theta$. The change in Y coordinate is $1 \times \sin\theta$. This can be represented in the following unit circle diagram:



The object has moved from $(x=0, y=0)$ at the center of the circle, to a new position on the edge of the circle located at: (x,y) where:

$$x = 0 + 1x\cos\theta \quad \text{and}$$
$$y = 0 + 1x\sin\theta$$

Source: <http://www.mathsisfun.com/geometry/images/circle-unit-sct.gif>

This situation is explained very clearly in Gary Rosenzweig's book: "Actionscript Game Programming University" page 228-233.

The other thing about this situation is that in Actionscript we might rotate an object 30 degrees to the right, but we need to convert these degrees into "radian" units before we can obtain the cosine and sine values. We need these Cos and Sin values so we can work out the latest X and Y coordinates as the object moves 1 unit in the θ direction.

So as per Gary's book, we setup for the car to rotate 5 degrees to the left (which is the negative direction), whenever the left arrow is pressed. We do a similar thing for the right arrow, but make the car rotate 5 degrees right. If the up arrow is pressed, we move the car forward:

```
// turn left or right or move car forward
public function moveCar(event:Event) {
```

Trigonometry in Car Driving Games with Flash Actionscript

```
if (leftArrow) {
    car.rotation -= 5;
}
if (rightArrow) {
    car.rotation += 5;
}
if (upArrow) {
    moveForward();
}
```

We then need to “Do the Math” to work out the value of the X and Y coordinates after each press of the arrow keys. Part of this calculation involves converting the degrees angle into units of “radians” by using the mathematical formula:

Radians = 2 x pi x (θ / 360) where pi can be set to 3.1416

We then find the Cosine and Sine values of this Radians value.

The following Actionscript code achieves this:

```
// calculate x and y speed and move car
public function moveForward() {
    var speed:Number = 5.0;
    var angle:Number = 2*Math.PI*(car.rotation/360);
    var dx:Number = speed*Math.cos(angle);
    var dy:Number = speed*Math.sin(angle);
    car.x += dx;
    car.y += dy;
}
```

And the all important end result is that we have a car that we can drive around our screen using the arrow keys 😊

