

# *Introduction to Groups*

*written by Dorothy Zeiser*

*This is a first draft of a worksheet for the beginning of Abstract Algebra  
(Needs more work)*

## ► **Outline**

1. Investigate the symmetries of a square visually.
2. Investigate the composition of two transformations visually.
3. Investigate the symmetries of a hexagon

## ▼ **Symmetries of a Square**

### ▼ **Define and plot a square**

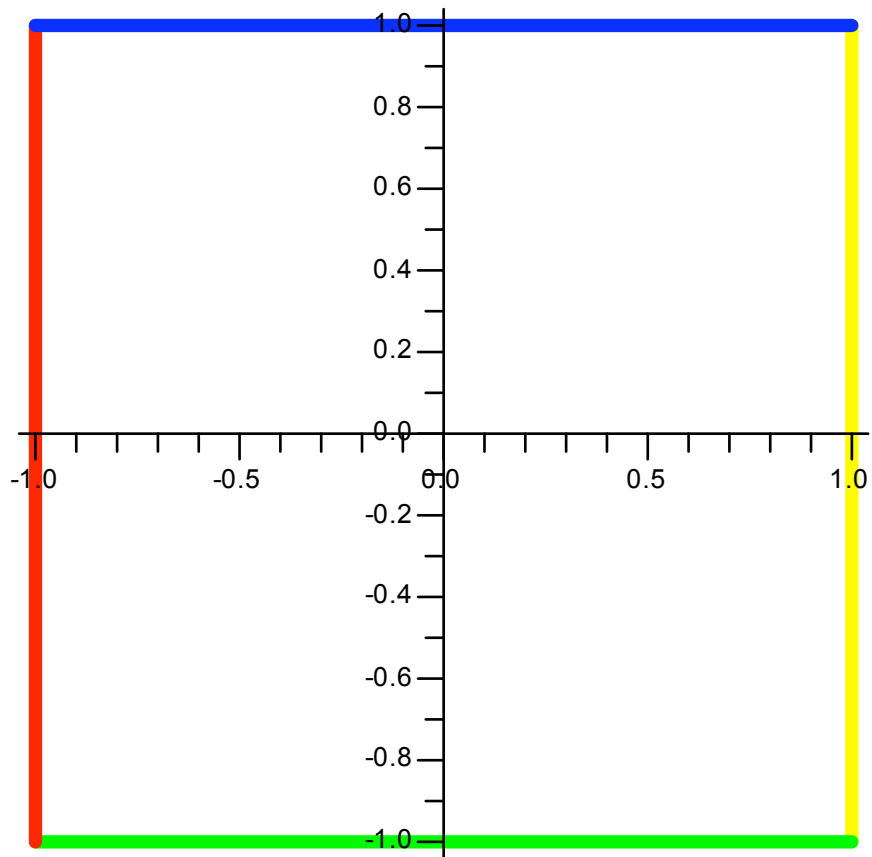
```
> restart : with (plots) : with (plottools) :  
Warning, the name changecoords has been redefined  
Warning, the assigned name arrow now has a global binding
```

The following define the sides of a square.

```
> L1 := line([1, 1], [-1, 1], color = blue, thickness = 5) :  
L2 := line([-1, 1], [-1, -1], color = red, thickness = 5) :  
L3 := line([-1, -1], [1, -1], color = green, thickness = 5) :  
L4 := line([1, -1], [1, 1], color = yellow, thickness = 5) :
```

Next we display the square.

```
> display(L1, L2, L3, L4);
```



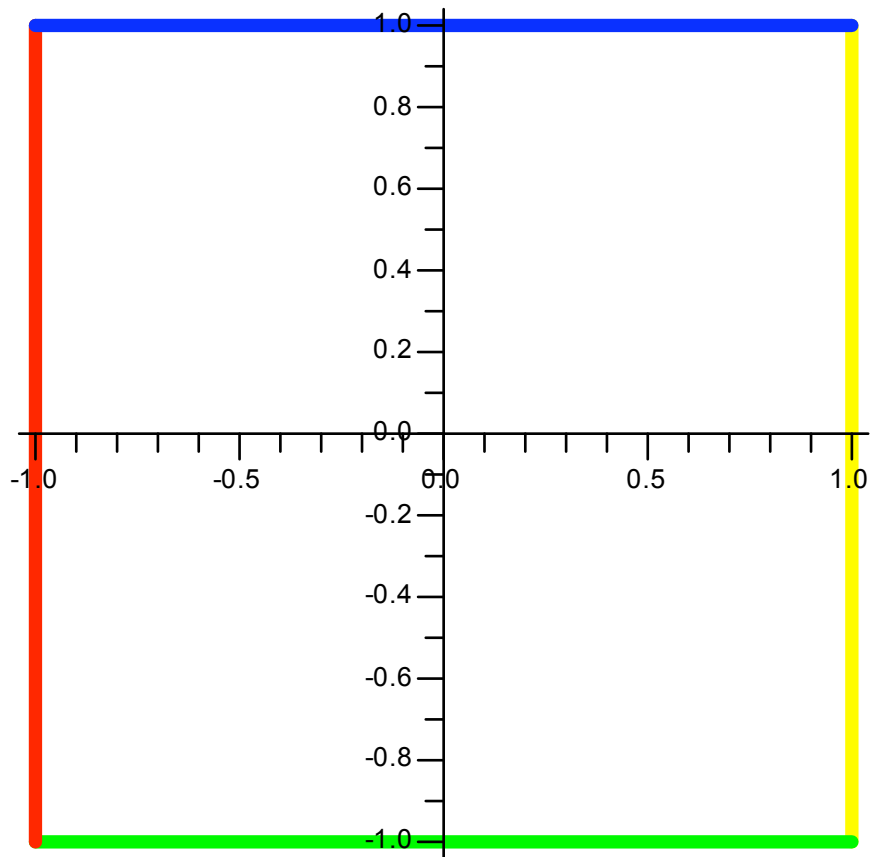
```
>
```

```
> square := display(L1, L2, L3, L4) :
```

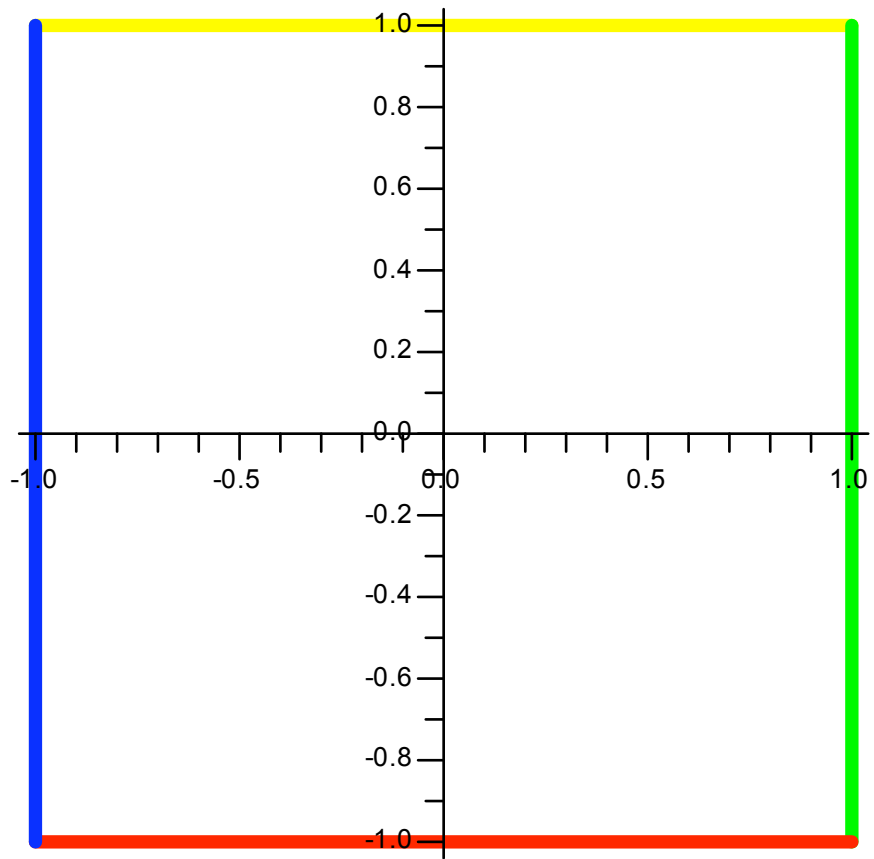
```
>
```

## ▼ Rotations

```
> R0 := rotate(square, 0, [0, 0]) :  
display(R0) ;
```

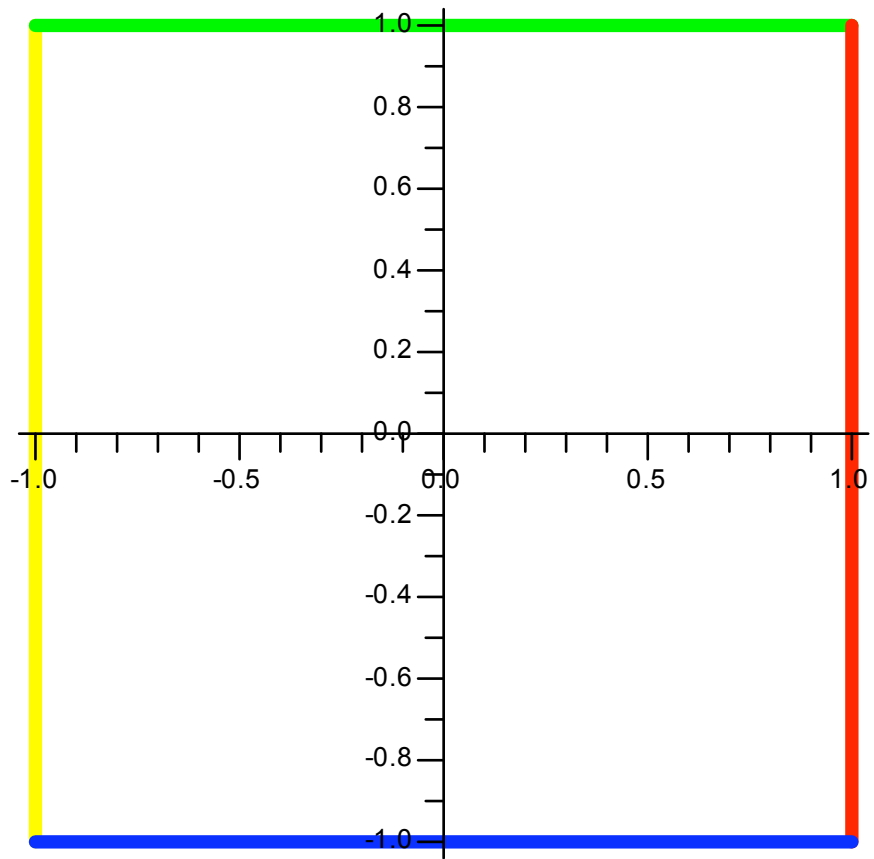


```
> R90 := rotate(square, Pi/2, [0,0]):  
display(R90);
```



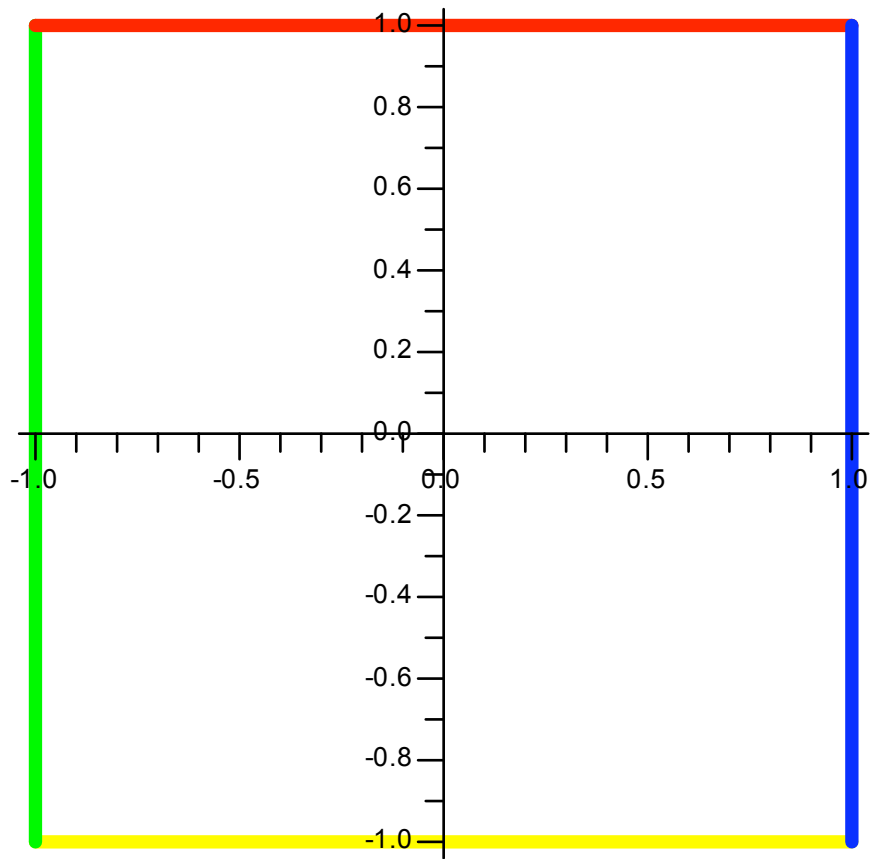
>

```
> R180 := rotate(square,Pi,[0,0]):  
display(R180);
```



>

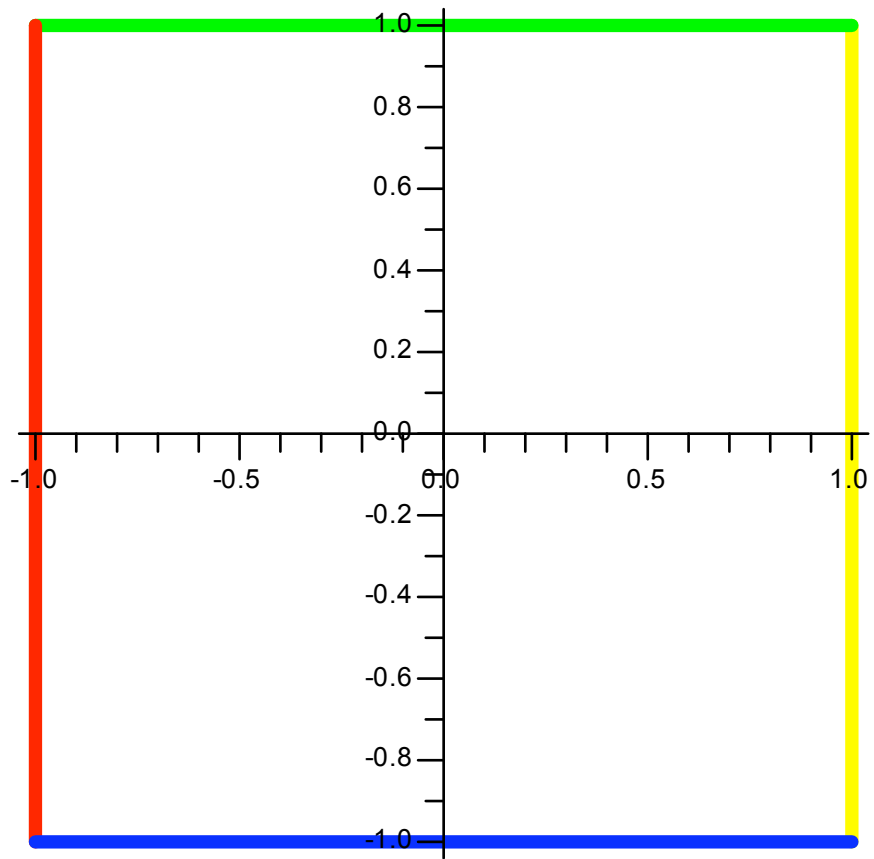
```
> R270 := rotate(square, 3*Pi/2, [0,0]):  
display(R270);
```



>

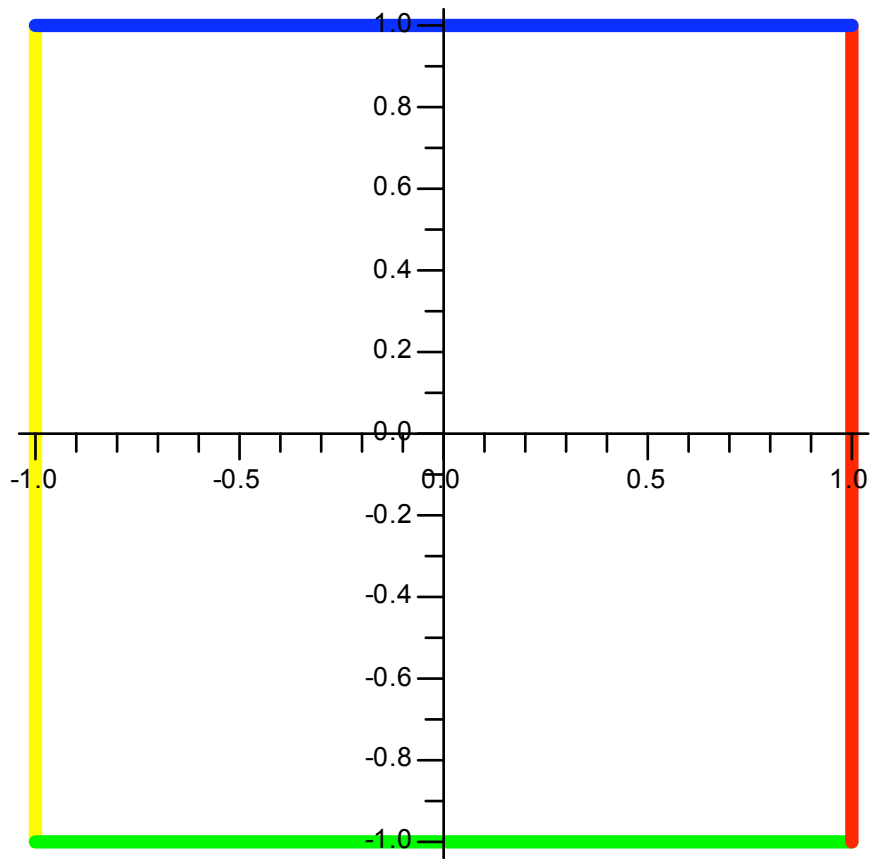
## ▼ Reflections

```
> H := reflect(square, [[-1,0],[1,0]]):  
display(H);
```



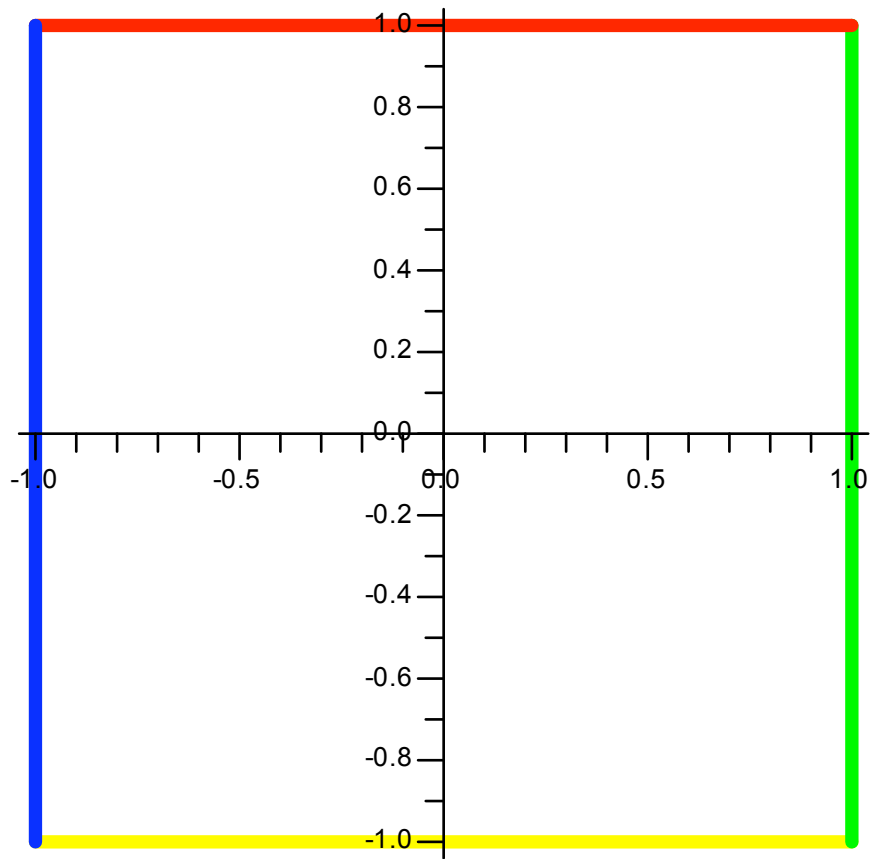
>

```
> v := reflect(square, [[0,1],[0,-1]]):  
display(v);
```



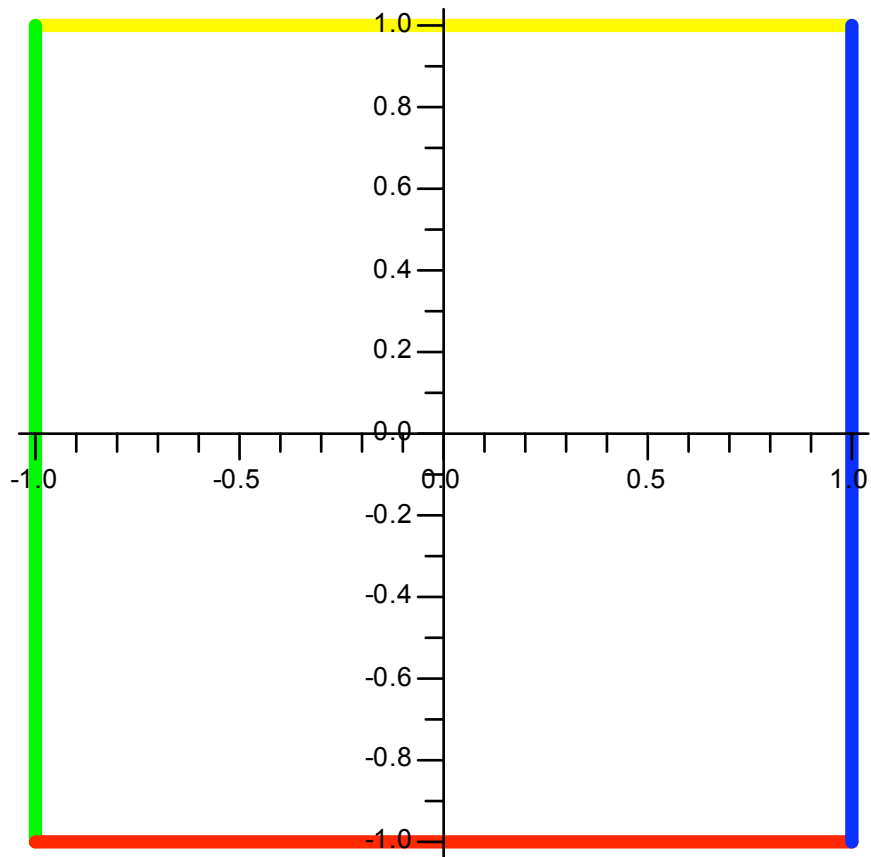
>

```
> MD := reflect(square, [[-1,1],[1,-1]]):  
display(MD);
```



>

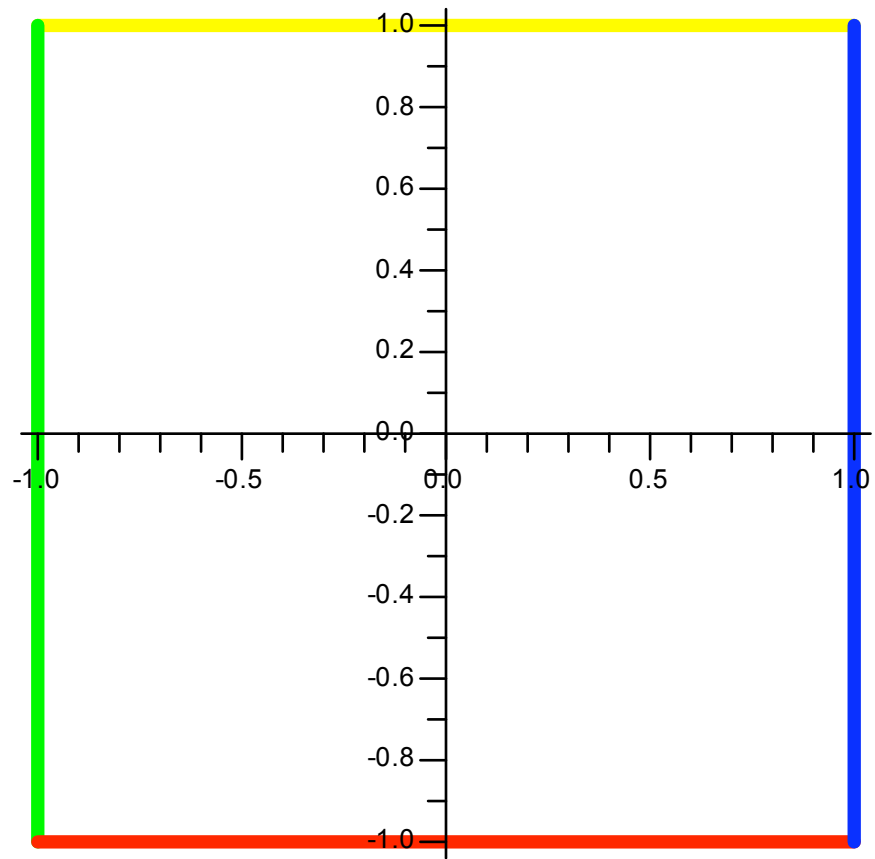
```
> SD := reflect(square, [[-1,-1],[1,1]]):  
display(SD);
```



>

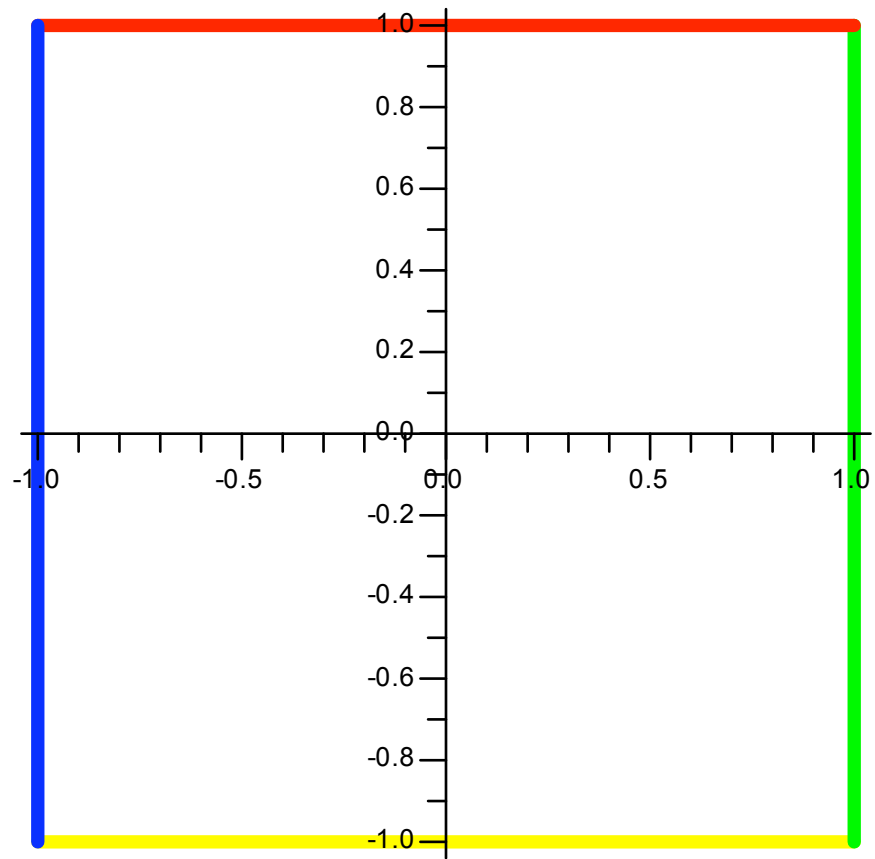
## ► Composition of Transformations

```
> R90H := rotate(H, Pi/2, [0,0]):  
display(R90H);
```



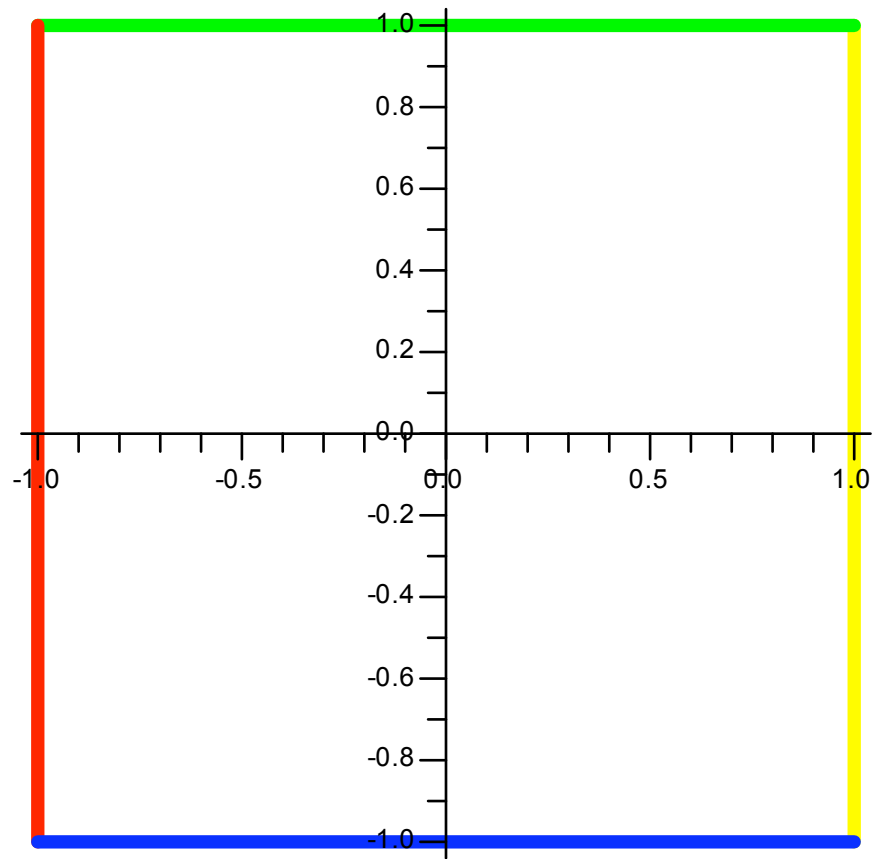
>

```
> R90V := rotate(V,Pi/2,[0,0]):  
display(R90V);
```



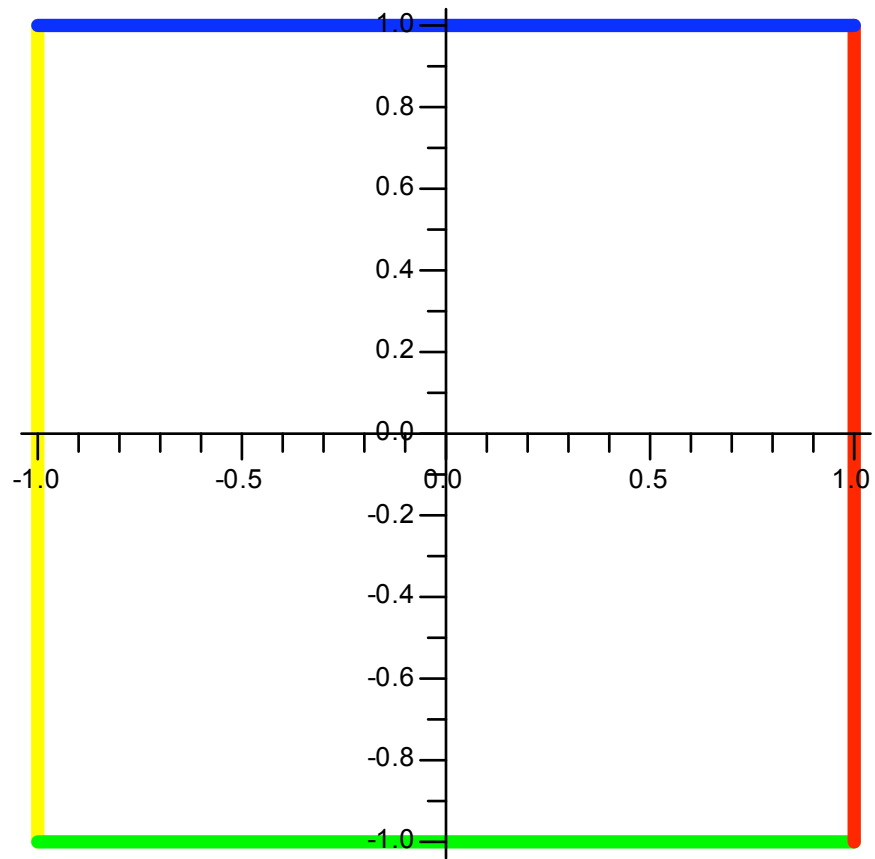
>

```
> R90MD := rotate(MD, Pi/2, [0,0]):  
display(R90MD);
```



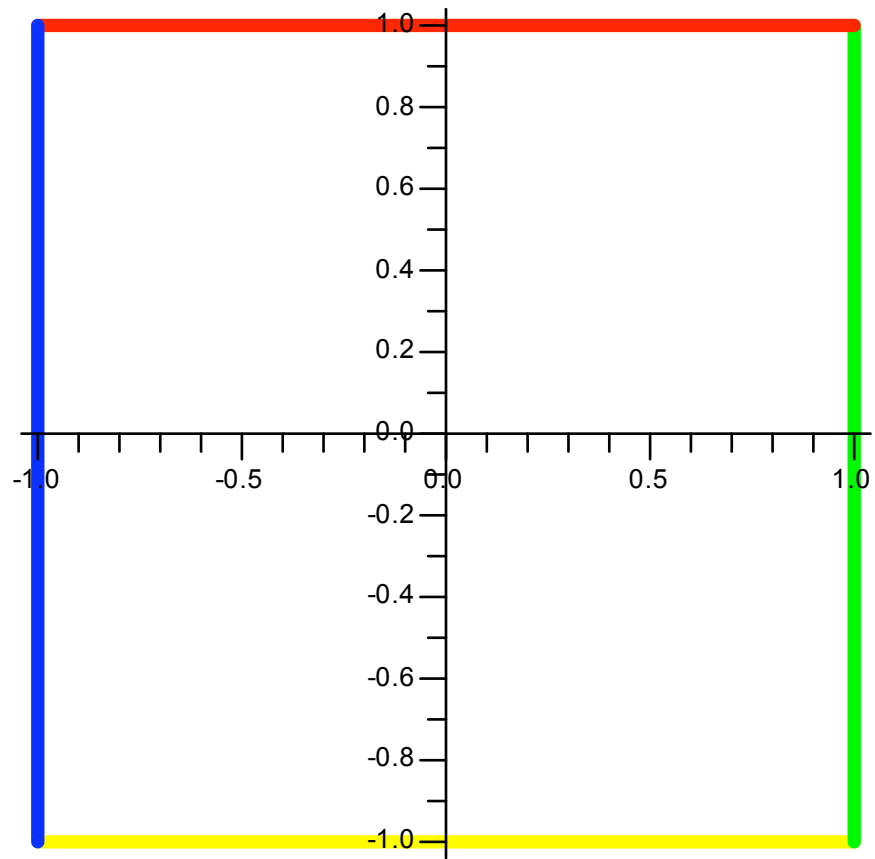
>

```
> R90SD := rotate(SD,Pi/2,[0,0]):  
display(R90SD);
```

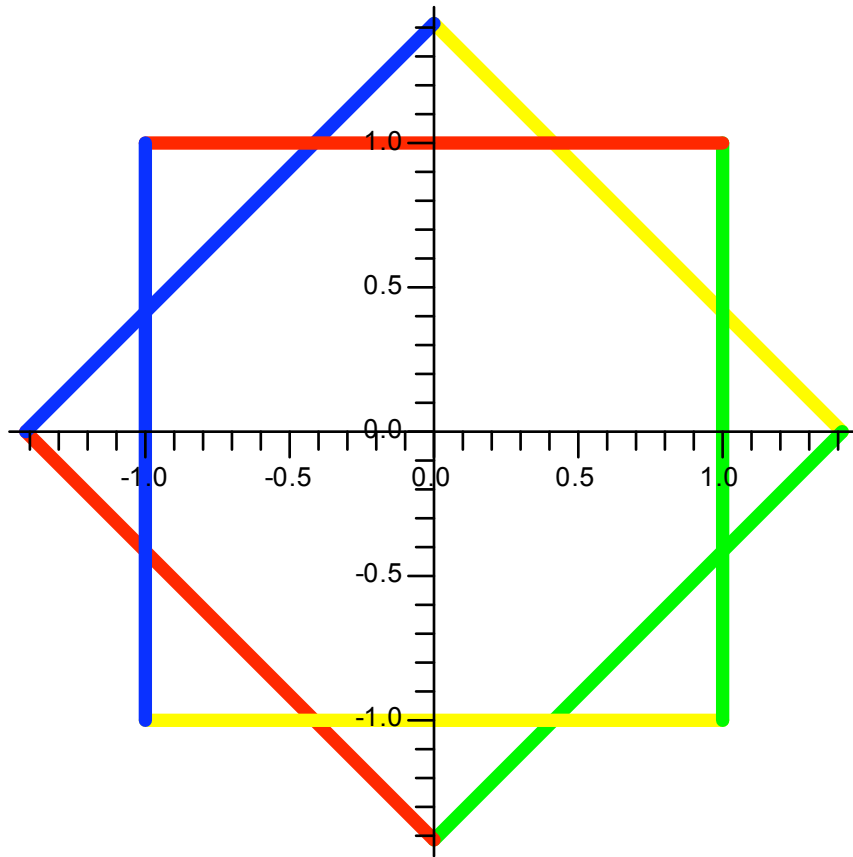


>

```
> HR90 := reflect(R90, [[-1,0],[1,0]]):  
display(HR90);
```



```
> R45 := rotate(square, Pi/4):  
display(HR90, R45);
```



## ▼ Assignment: Symmetries of a Regular Hexagon

### ▼ Define and plot a regular hexagon

```
> restart : with(plots) : with(plottools) :
Warning, the name changecoords has been redefined
Warning, the assigned name arrow now has a global binding
```

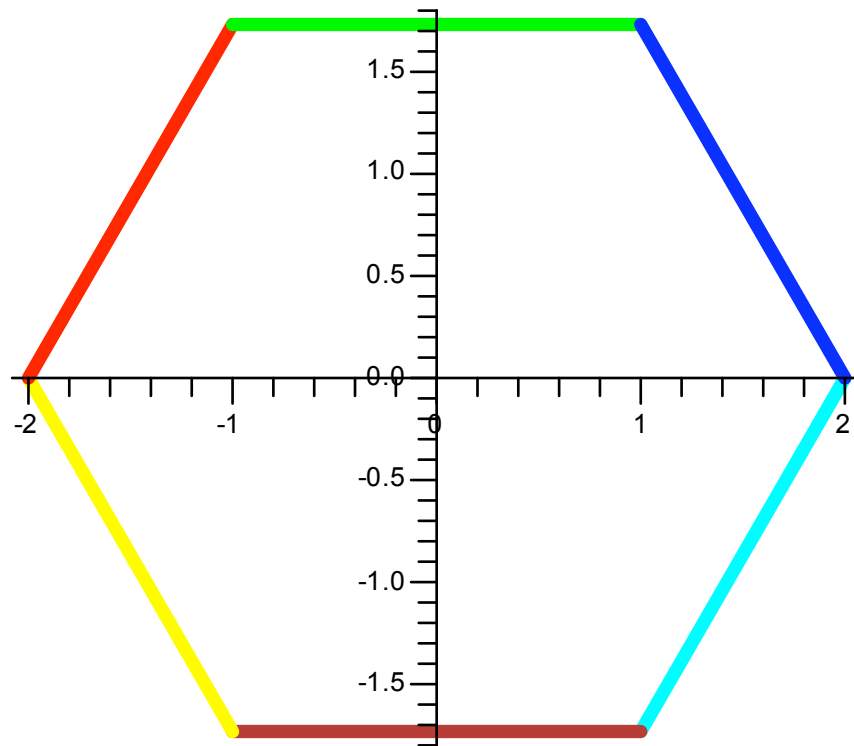
The following define the sides of a hexagon.

```
> H1 := line([2,0],[1,sqrt(3)],color=blue,thickness=5):
H2 := line([1,sqrt(3)],[-1,sqrt(3)],color=green,thickness=5):
H3 := line([-1,sqrt(3)],[-2,0],color=red,thickness=5):
H4 := line([-2,0],[-1,-sqrt(3)],color=yellow,thickness=5):
```

```
H5 := line([-1,-sqrt(3)],[1,-sqrt(3)],color=brown,thickness=5):  
H6 := line([1,-sqrt(3)],[2,0],color=cyan,thickness=5):
```

The following displays the hexagon.

```
> display([H1,H2,H3,H4,H5,H6],scaling=constrained);
```



```
>
```

The following will enable you to work with the hexagon as an object.

```
> hexagon := display([H1,H2,H3,H4,H5,H6]):
```

```
>
```

## ▼ Rotations

## ▼ Reflections

## ▼ Rotations

**1. What is the angle measure of each rotation?**

*Type your answer here →*

**2. What is the radian measure of each angle of rotation?**

*Type your answers here →*

**3. Below copy— paste— edit maple code from previous example.**

***Generate two different rotations.***

***Generate the product of these.***

## ▼ Reflections (later)