3.016 Problem Set #4, 2010

I4-1

First express the matrix in matrix format:

```
matrixI41 =
  {
   {1, 4, -2},
   {4, 1, 2},
   {-2, 2, -2}};
MatrixForm[matrixI41]
```

$$\begin{pmatrix} 1 & 4 & -2 \\ 4 & 1 & 2 \\ -2 & 2 & -2 \end{pmatrix}$$

Find out determinant of the matrix :

```
Det[matrixI41]
```

-10

Note that the determinant is not zero.

Now define the rotation matrix with the rotation angle θ around x-axis:

```
rotationX = RotationMatrix[θ, {1, 0, 0}];
MatrixForm[rotationX]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \text{Cos}[\theta] & -\text{Sin}[\theta] \\ 0 & \text{Sin}[\theta] & \text{Cos}[\theta] \end{pmatrix}$$

Now define the rotation matrix with the rotation angle θ around z-axis:

```
rotationZ = RotationMatrix[θ, {0, 0, 1}];
MatrixForm[rotationZ]
```

$$\begin{pmatrix} \text{Cos}[\theta] & -\text{Sin}[\theta] & 0 \\ \text{Sin}[\theta] & \text{Cos}[\theta] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- **The first operation is the first to multiply, so the second operation "left multiplies" the first operation.**

```
firstXthenZ = rotationZ.rotationX;
MatrixForm[firstXthenZ]
```

```
( Cos[θ]  -Cos[θ] Sin[θ]     Sin[θ]^2
  Sin[θ]     Cos[θ]^2     -Cos[θ] Sin[θ]
    0         Sin[θ]          Cos[θ]     )
```

The above is the overall transformation matrix for the first rotation sequence.
Now find out the final new matrix after the first rotation sequence:

```
newMatrixXthenZ =
  Simplify[Inverse[firstXthenZ].matrixI41.firstXthenZ];
MatrixForm[newMatrixXthenZ]
```

```
(                 1 + 8 Cos[θ] Sin[θ]                                   1 + 2 Cos[θ] - Cos[2 θ] + 2 Cos[3 θ] - Sin[2 θ
  1 + 2 Cos[θ] - Cos[2 θ] + 2 Cos[3 θ] - Sin[2 θ]    -1/2 + Cos[θ] + 3/2 Cos[2 θ] - Cos[3 θ] + Sin[θ] - 2 Sin[2 θ] + S
 -1 - Cos[2 θ] + 2 Sin[θ] + Sin[2 θ] - 2 Sin[3 θ]      1 + Cos[θ] + Cos[3 θ] - Cos[4 θ] - Sin[θ] - 3 Cos[θ] Sin
```

Check the determinant, and simplify it:

```
Det[newMatrixXthenZ]
Simplify[Det[newMatrixXthenZ]]
```

```
-(1 + Cos[θ] + Cos[3 θ] - Cos[4 θ] - Sin[θ] - 3 Cos[θ] Sin[θ] + Sin[3 θ])
  (-(1 + 2 Cos[θ] - Cos[2 θ] + 2 Cos[3 θ] - Sin[2 θ]) (-1 - Cos[2 θ] + 2 Sin[θ] + Sin[2 θ] - 2 Sin[3 θ]) +
    (1 + 8 Cos[θ] Sin[θ]) (1 + Cos[θ] + Cos[3 θ] - Cos[4 θ] - Sin[θ] - 3 Cos[θ] Sin[θ] + Sin[3 θ])) +
 (-1 - Cos[2 θ] + 2 Sin[θ] + Sin[2 θ] - 2 Sin[3 θ]) ((1 + 2 Cos[θ] - Cos[2 θ] + 2 Cos[3 θ] - Sin[2 θ])
     (1 + Cos[θ] + Cos[3 θ] - Cos[4 θ] - Sin[θ] - 3 Cos[θ] Sin[θ] + Sin[3 θ]) -
    (-1 - Cos[2 θ] + 2 Sin[θ] + Sin[2 θ] - 2 Sin[3 θ])
     (-1/2 + Cos[θ] + 3/2 Cos[2 θ] - Cos[3 θ] + Sin[θ] - 2 Sin[2 θ] + Sin[3 θ] - Sin[4 θ])) +
 (-(1 + 2 Cos[θ] - Cos[2 θ] + 2 Cos[3 θ] - Sin[2 θ])^2 + (1 + 8 Cos[θ] Sin[θ])
     (-1/2 + Cos[θ] + 3/2 Cos[2 θ] - Cos[3 θ] + Sin[θ] - 2 Sin[2 θ] + Sin[3 θ] - Sin[4 θ]))
  (-1/2 - Cos[θ] - 3/2 Cos[2 θ] + Cos[3 θ] - Sin[θ] - 2 Sin[2 θ] - Sin[3 θ] + Sin[4 θ])
```

```
-10
```

The determinant is still -10, the same as the original matrix!

Now do the rotation with a difference order, first θ around z-axis and then θ around x-axis:

```
firstZthenX = rotationZ.rotationZ;
MatrixForm[firstZthenX]
```

$$\begin{pmatrix} \mathrm{Cos}[\theta]^2 - \mathrm{Sin}[\theta]^2 & -2\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta] & 0 \\ 2\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta] & \mathrm{Cos}[\theta]^2 - \mathrm{Sin}[\theta]^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

After getting the above transformation matrix, find out the final new matrix after the second rotation sequence:

```
newMatrixZthenX =
  Simplify[Inverse[firstZthenX].matrixI41.firstZthenX];
MatrixForm[newMatrixZthenX]
```

$$\begin{pmatrix} 1 + 4\,\mathrm{Sin}[4\,\theta] & 4\,\mathrm{Cos}[4\,\theta] & -2\,\mathrm{Cos}[2\,\theta] + 2\,\mathrm{Sin}[2\,\theta] \\ 4\,\mathrm{Cos}[4\,\theta] & 1 - 4\,\mathrm{Sin}[4\,\theta] & 2\,(\mathrm{Cos}[2\,\theta] + \mathrm{Sin}[2\,\theta]) \\ -2\,\mathrm{Cos}[2\,\theta] + 2\,\mathrm{Sin}[2\,\theta] & 2\,(\mathrm{Cos}[2\,\theta] + \mathrm{Sin}[2\,\theta]) & -2 \end{pmatrix}$$

Note that after changing the sequence of rotation, the final transformed matrix is also different!
Now find out the determnant:

```
Det[newMatrixZthenX]
Simplify[Det[newMatrixZthenX]]
```

$$-2 - 8\,\mathrm{Cos}[2\,\theta]^2 - 32\,\mathrm{Cos}[2\,\theta]^2\,\mathrm{Cos}[4\,\theta] + 32\,\mathrm{Cos}[4\,\theta]^2 - 8\,\mathrm{Sin}[2\,\theta]^2 + 32\,\mathrm{Cos}[4\,\theta]\,\mathrm{Sin}[2\,\theta]^2 - 64\,\mathrm{Cos}[2\,\theta]\,\mathrm{Sin}[2\,\theta]\,\mathrm{Sin}[4\,\theta] + 32\,\mathrm{Sin}[4\,\theta]^2$$

$$-10$$

- **The determinants are unchanged after two different rotation sequences, the determinants are invarient.**

I4-2

Here is some Mathematica code that takes a vector of numbers for red, green, and blue and draws a colored circle:

```
rgbCircle[{r_?NumberQ, g_?NumberQ, b_?NumberQ}] := Module[
  {maxval = Abs[ Max[{r, g, b}]], rs, bs, gs},
  {rs, gs, bs} = Abs[{r, g, b}] / maxval;
  Graphics[{RGBColor[rs, gs, bs], Disk[]}]
 ]
```

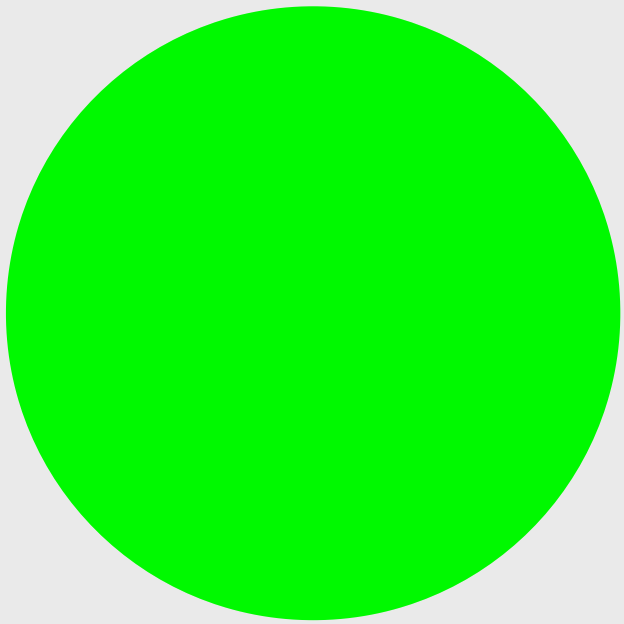
```
rgbCircle[{12, 4, 1}]
```

This is a red circle:

```
rgbCircle[{1, 0, 0}]
```

This is a green circle:

```
rgbCircle[{0, 1, 0}]
```



This is a blue circle:

```
rgbCircle[{0, 0, 1}]
```

- **Purple is half blue and half red, Orange is half red and half green, and Cyan is half green and half blue. First define the matrix that converts a vector given as (r, g, b) to a vector given as purple, orange, cyan (p, o, c).**

```
rgbTopoc = {
   {1 / 2, 0, 1 / 2},
   {1 / 2, 1 / 2, 0},
   {0, 1 / 2, 1 / 2}
  };
MatrixForm[rgbTopoc]
```

$$\begin{pmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

```
Det[rgbTopoc]
```

$$\frac{1}{4}$$

Determinant is not zero. Find out the inverse of the above matrix:

```
pocTorgb = Inverse[rgbTopoc];
MatrixForm[pocTorgb]
```

$$\begin{pmatrix} 1 & 1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & 1 \end{pmatrix}$$

```
pocTorgb.{1, 0, 0}
```

{1, -1, 1}

```
pocCircle[{p_, o_, c_}] := rgbCircle[pocTorgb.{p, o, c}]
```

```
pocCircle[{1, 0, 0}]
```

```
pocCircle[{1, 1, 3}]
```

- **This didn't work as I expected because of the normalization that I picked, I will rewrite the function to use a unit normal for any color:**

```
rgbCircle[{r_?NumberQ, g_?NumberQ, b_?NumberQ}] := Module[
  {rs, bs, gs},
  {rs, gs, bs} = Abs[Normalize[{r, g, b}]];
  Graphics[{RGBColor[rs, gs, bs], Disk[]}]
 ]
```

```
rgbCircle[{1, 0, 0}]
```

```
pocCircle[{p_, o_, c_}] := rgbCircle[pocTorgb.{p, o, c}]
```

```
pocTorgb.{0, 1, 0}
```

```
{1, 1, -1}
```

```
pocCircle[{0, 1, 0}]
```

- **This still doesn't work... I think it is because everything must be mapped to the positive octant. This was a badly posed problem---my fault. Students will get credit for attempting the solution.**

I4-3

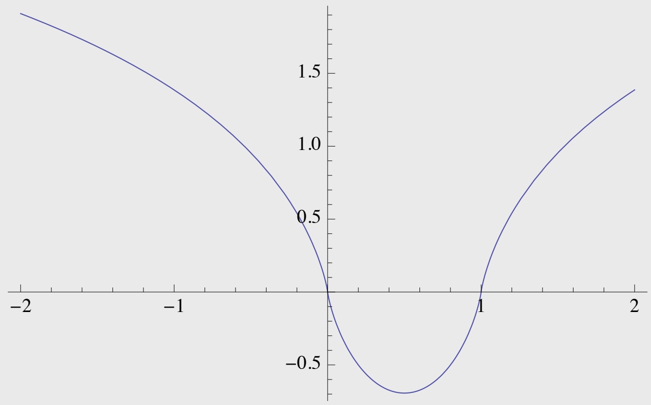
First define desired function:

```
blotz = z Log[z] + (1 - z) Log[1 - z]
```

```
(1 - z) Log[1 - z] + z Log[z]
```

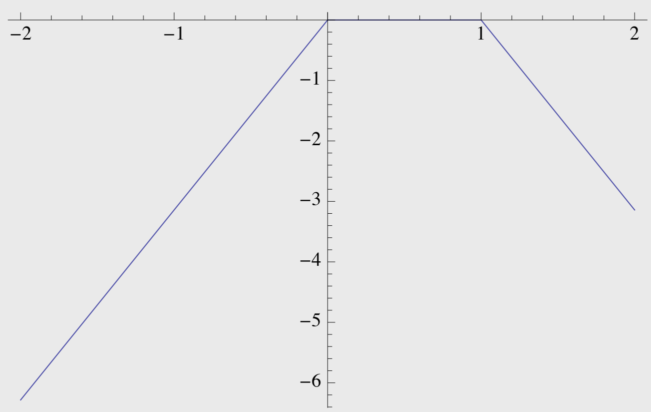
Plot the real part of the function:

```
Plot[Re[blotz], {z, -2, 2}]
```



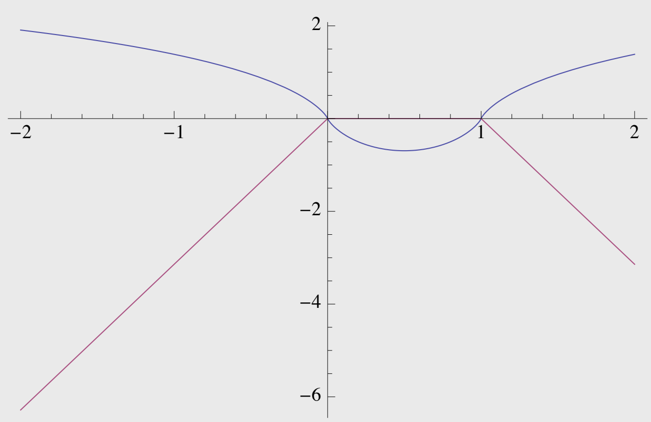
Plot the imaginary part of the function:

```
Plot[Im[blotz], {z, -2, 2}]
```



Plot them together:

```
Plot[{Re[blotz], Im[blotz]}, {z, -2, 2}]
```



Function is complex outside of (0,1). Imaginary part increases or decreases linearly.

- **Now taking the first and second derivatives:**

```
dblotz = D[blotz, z]
```

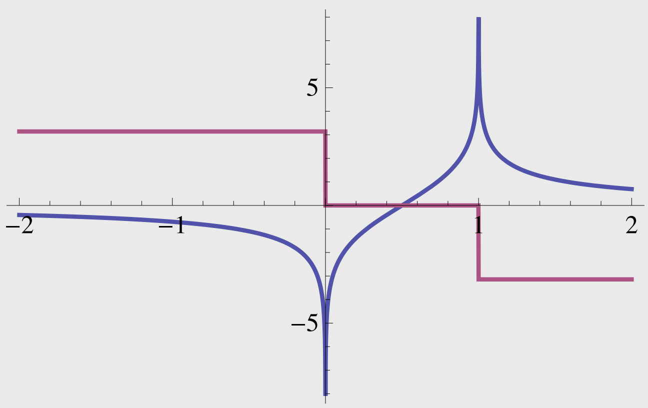
```
-Log[1 - z] + Log[z]
```

```
ddblotz = D[blotz, {z, 2}]
```

$$\frac{1}{1-z} + \frac{1}{z}$$

Now plot the real and imaginary parts of the first derivative:

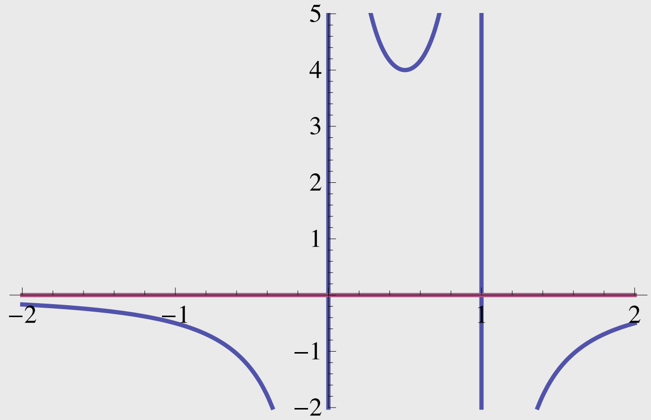
```
Plot[{Re[dblotz], Im[dblotz]}, {z, -2, 2}]
```



- **Thus, the first derivative produces three segments of constant imaginary part. Note that this function has an infinite derivative at 0 and 1, although the function itself is finite.**

Now plot the real and imaginary parts of the second derivative:

```
Plot[{Re[ddblotz], Im[ddblotz]}, {z, -2, 2}, PlotRange → {-2, 5}]
```



- **Thus, the second derivative produces a function that is real everywhere, except at the singularities at 0 and 1.**

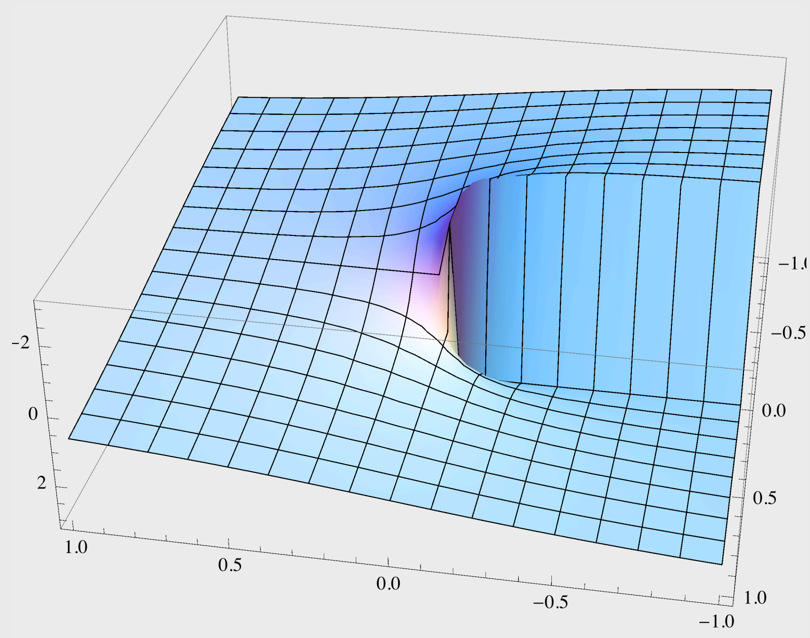
I4-4

- **First define the plotting function with a variable range...**

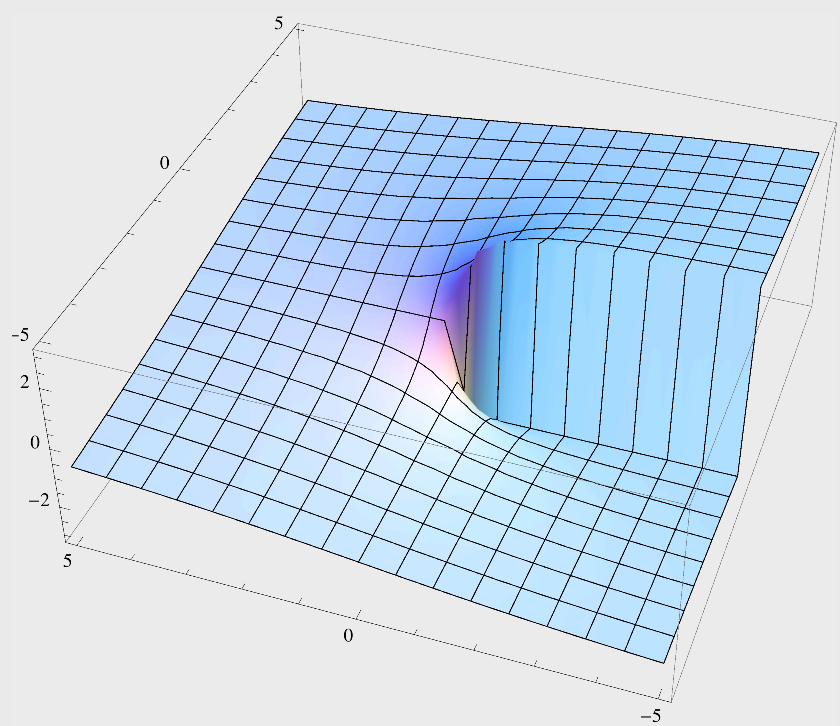
```
thePlot[range_] :=
 Plot3D[ArcTan[x, y], {y, -range, range}, {x, -range, range}]
```

Plot the function with different ranges:

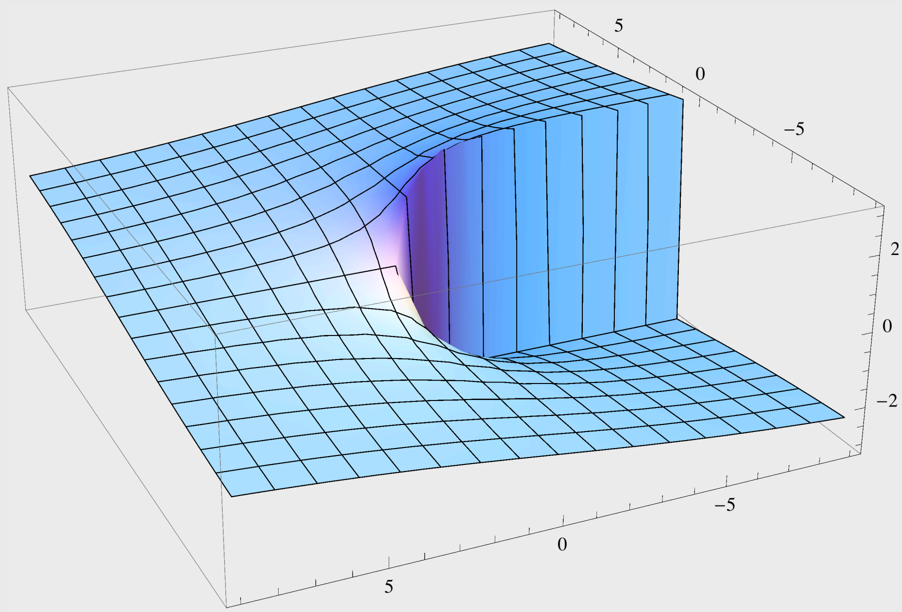
```
thePlot[1]
```




```
thePlot[5]
```



```
thePlot[9]
```



- **This looks like a screw dislocation...**