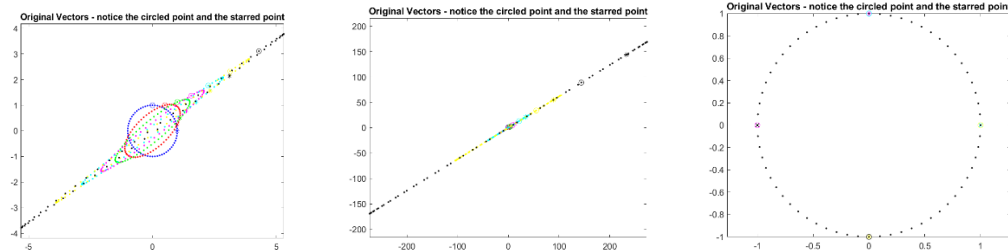


1.

1. Points get stretched towards $y=x$
2. Points get stretched towards $y=x$, but shallower and to a greater degree than figure 1
3. Points get stretched towards $y=x$, but steeper and to a greater degree than figure 1, almost inverse of figure 2
4. All points rotate clockwise 90 degrees around origin
5. All points reflect across $y=x$
6. All points rotate clockwise about 4 degrees around origin

2.



i.

ii.

1. Points stretch out to $(-5, -3.5)$ and $(5, 3.5)$
2. Points stretch out to $(-250, -150)$ and $(250, 150)$, resembles a straight line
3. Points do 6 90 degree clockwise rotations around the origin

iii. Points do 15 ~4 degree clockwise rotations around the origin

3.

```
if size(A)~= [2,2]    conditional IF statement comparing the size of matrix A to a 2x2 matrix
fprintf('A must be a 2 x 2 matrix');  If false, print test to the window
return;              If false, return nothing and end the program
end                  End of IF statement
```

```
color = str2mat('b ','r ','g ','m ','c ','y ','k ');    assign 'color' variable to typecast symbols
cc1 = str2mat('b *','r *','g *','m *','c *','y *','k *');  assign 'cc1' variable to typecast symbols
cc2 = str2mat('b o','r o','g o','m o','c o','y o','k o');  assign 'cc2' variable to typecast symbols
```

```
% Creating unit vectors
figure;    create a blank figure
```

```
it=0; NN=64;    assign variables 'it' and 'NN' to 0 and 64
if it == 0      conditional IF statement comparing 'it' to 0
V=zeros(2,NN);  if true, assign a 2x'NN' vector of all zeroes to matrix V
V = [cos(2*pi*[0:NN-1]/NN); sin(2*pi*[0:NN-1]/NN)];    if true, assign matrix V the values
cos(2*pi*[0:NN-1]/NN) and sin(2*pi*[0:NN-1]/NN)
Vorig=V;    assign the matrix Vorig the contents of matrix V
end        End of IF statement
```

plot(V(1,:),V(2,:), 'b.') plot part of matrix V

hold on start holding

plot(V(1,1),V(2,1), 'b*') plot part of matrix V

plot(V(1,NN/4+1),V(2,NN/4+1), 'bo') plot part of matrix V

title('Original Vectors - notice the circled point and the starred point') add title to figure

fprintf('Hit any key to continue.') notify user with text prompt

pause wait for keyboard input

for kk=1:NumIters for loop going from when kk=1 and continuing while kk is less than NumIters

V=A*V; assign matrix A with the product of matrix A and V

plot(V(1,:),V(2,:),color(rem(kk,7)+1,:)) plot points from matrix

plot(V(1,1),V(2,1),cc1(rem(kk,7)+1,:)) plot points from matrix

plot(V(1,NN/4+1),V(2,NN/4+1),cc2(rem(kk,7)+1,:)) plot points from matrix

axis('equal') set axis to 'equal'

fprintf('Hit any key to continue.') notify user with text prompt

pause wait for keyboard input

end end of for loop

hold off stop holding

Aout=V; assign matrix Aout the contents of matrix V