QUESTION-Find the equation of the tangent to the circle, at the point "B"-

$$\mathsf{B} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

whose centre is the point of intersection of the straight lines

$$\begin{pmatrix} 2 & 1 \end{pmatrix} \times \begin{pmatrix} x \\ y \end{pmatrix} = (3) \tag{1}$$

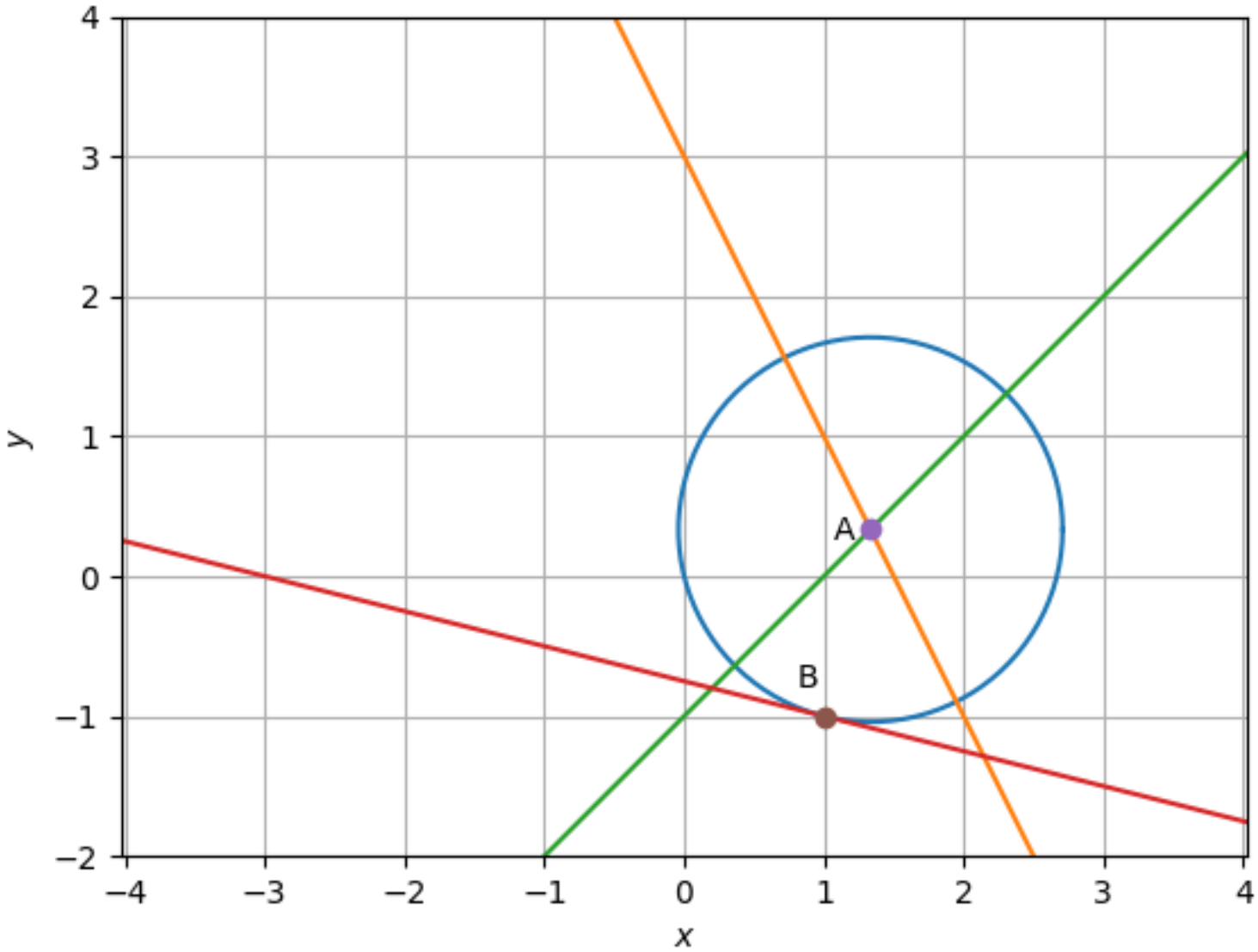
$$\begin{pmatrix} 1 & -1 \end{pmatrix} \times \begin{pmatrix} x \\ y \end{pmatrix} = (1) \tag{2}$$

Solution- the intersection of both lines gives us the centre of circle
point of intersection can be computed as

$$\mathsf{P} = \begin{pmatrix} 2 & 1 \\ 1 & -1 \end{pmatrix} \tag{3}$$

$$\begin{pmatrix} 2 & 1 \\ 1 & -1 \end{pmatrix} \times \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \end{pmatrix} \tag{4}$$

solution for above equation gives us centre "A"
A is (4/3,1/3)
normal vector of tangent is along line joining A and B
hence
the equation of tangent at B can be written as
$(\mathbf{B} - \mathbf{A})^{\intercal}(\mathbf{x} - \mathbf{B}) = 0$
following is the plot:



using the code above figure can be obtained and tangent equation can be computed-

```
import numpy as np
import matplotlib.pyplot as plt
a=np.array([2,1])
b=np.array([1,-1])

N1=np.vstack((a,b))
p1=np.array([3,1])
A=np.matmul(np.linalg.inv(N1),p1)

B=np.array([1,-1])
n=np.matmul(np.array([[0,-1],[1,0]]),B-A)
r=np.linalg.norm(B-A)
len=100

print(n)



x1=np.zeros((2,len))
x2=np.zeros((2,len))
```

```
x3=np.zeros((2,len))
x4=np.zeros((2,len))
lam1=np.linspace(0,2*np.pi,len)
lam2=np.linspace(-10,10,len)


for i in range(len):
        temp1= np.array([A[0]+r*np.cos(lam1[i]),A[1]
        x1[:,i]=temp1.T
for i in range(len):
        temp1= np.array([lam2[i],3-2*(lam2[i])])
        x2[:,i]=temp1.T
for i in range(len):
        temp1= np.array([lam2[i],lam2[i]-1])
        x3[:,i]=temp1.T
for i in range(len):
        temp1=B+lam2[i]*n
        x4[:,i]=temp1.T
plt.plot(x1[0,:],x1[1,:])
```

```
plt.axis('equal')
plt.plot(x2[0,:],x2[1,:])
plt.axis('equal')
plt.plot(x3[0,:],x3[1,:])
plt.axis('equal')
plt.plot(x4[0,:],x4[1,:])
plt.axis('equal')
plt.plot(A[0],A[1],'o')
plt.text(A[0]*(1-0.2),A[1]*(1-0.2),'A')
plt.plot(B[0],B[1],'o')
plt.text(B[0]*(1-0.2),B[1]*(1-0.2),'B')
plt.xlabel('$x$')
plt.ylabel('$y$')
plt.xlim(-3,5)
plt.ylim(-3,5)


plt.grid()
plt.show()
```