# Assignment 3

Teja Vardhan Shannu

August 2024

# Problem Statement

A line intersects the Y-axis and the X-axis at the points $P\,(0, b)$ and $Q\,(c, 0)$ respectively. If $(2, -5)$ is the midpoint of $PQ$, find the coordinates of $P$ and $Q$.

# Problem Data

| **Point** | **Coordinates** |
|---|---|
| $P$ | $\begin{pmatrix} 0 \\ b \end{pmatrix}$ |
| $Q$ | $\begin{pmatrix} c \\ 0 \end{pmatrix}$ |
| $M$ | $\begin{pmatrix} 2 \\ -5 \end{pmatrix}$ |

# Midpoint Formula

Let the coordinates of points $P$ and $Q$ be:

$$\mathbf{P} = \begin{pmatrix} 0 \\ b \end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix} c \\ 0 \end{pmatrix}$$

The midpoint $M$ is given by:

$$\mathbf{M} = \begin{pmatrix} 2 \\ -5 \end{pmatrix}$$

The midpoint formula states:

$$\mathbf{M} = \frac{1}{2}(\mathbf{P} + \mathbf{Q})$$

# Applying the Midpoint Formula

Substitute **P**, **Q**, and **M**:

$$\frac{1}{2}\left(\begin{pmatrix}0\\b\end{pmatrix}+\begin{pmatrix}c\\0\end{pmatrix}\right)=\begin{pmatrix}2\\-5\end{pmatrix}$$

Simplify to get:

$$\frac{1}{2}\begin{pmatrix}c\\b\end{pmatrix}=\begin{pmatrix}2\\-5\end{pmatrix}$$

Multiplying both sides by 2:

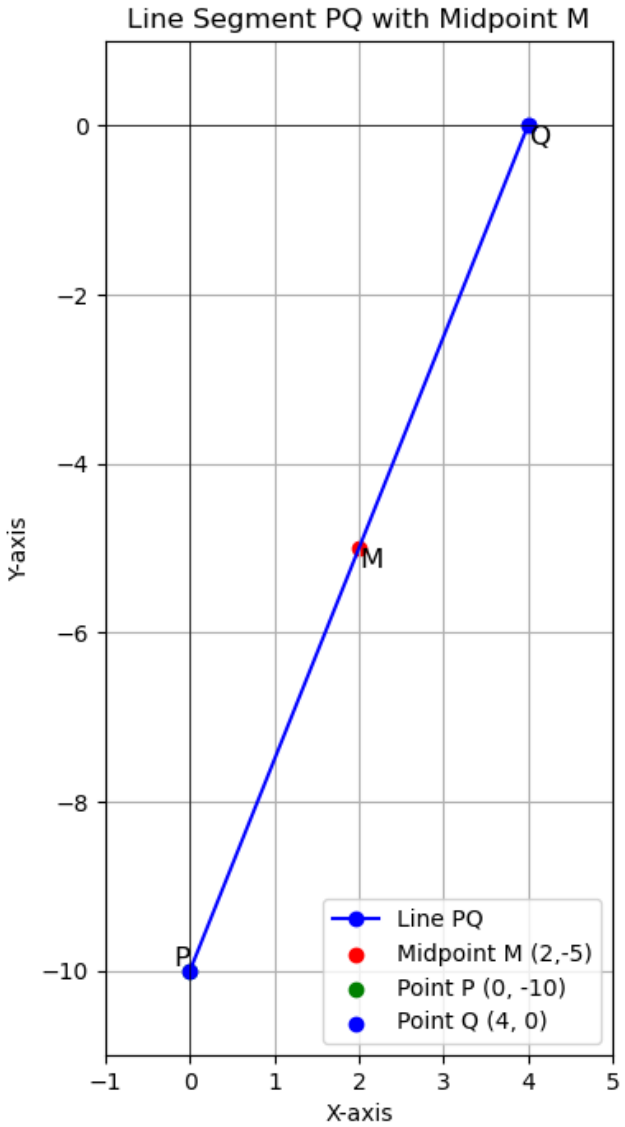
$$\begin{pmatrix}c\\b\end{pmatrix}=\begin{pmatrix}4\\-10\end{pmatrix}$$

# Final Answer

Thus, the coordinates of $P$ and $Q$ are:

$$\mathbf{P} = \begin{pmatrix} 0 \\ -10 \end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix} 4 \\ 0 \end{pmatrix}$$

# Graphical Representation



# C Code

```c
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <sys/socket.h>
#include <netinet/in.h>

#include "libs/matfun.h"
#include "libs/geofun.h"
int main ()  {
    double **P,**Q,**M;
    P=createMat(2,1);
    Q=createMat(2,1);
    M=createMat(2,1);
    P[0][0]=0;
    Q[1][0]=0;
    M[0][0]=2;
    M[1][0]=-5;
```

```
    P[1][0]=2*(M[1][0]);
    Q[0][0]=2*(M[0][0]);

    FILE *file=fopen("output.dat","w");
    if (file == NULL)  {
        printf("Error opening file!\n");
        return 1;
    }

    fprintf(file,"P Q\n");
    fprintf(file,"%.2f %.2f\n",P[1][0],Q[0][0]);
    fclose(file);
    freeMat(P,2);
    freeMat(Q,2);
    freeMat(M,2);

    return 0;
}
```

# python Code

```
#Code by GVV Sharma
#September 12, 2023
#Revised July 21, 2024
#released under GNU GPL
#Rank
import sys                  #for path to external scripts
sys.path.insert
(0, '/home/teja-vardhan/Desktop/matgeo/codes/CoordGeo')
#path to my scripts
import numpy as np
import numpy.linalg as LA
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

#local imports
from line.funcs import *
from triangle.funcs import *
from conics.funcs import circ_gen
```

```
#if using termux
import subprocess
import shlex
#end if
data = np.genfromtxt
('output.dat', delimiter=' ', names=True)
x = data['P']
y = data['Q']

#Given points
P = np.array(([0,x])).reshape(-1,1)
Q = np.array(([y,0])).reshape(-1,1)
M = np.array(([2,-5])).reshape(-1,1)
#Print rank
print
(LA.matrix_rank(np.block([Q-P,M-P])))
#Generating all lines
x_PQ = line_gen(P,Q)
```

```
#Plotting all lines
plt.plot(x_PQ[0,:],x_PQ[1,:],label='$PQ$')

#Labeling the coordinates
tri_coords = np.block([[P,Q,M]])
plt.scatter(tri_coords[0,:], tri_coords[1,:])
vert_labels = ['P','Q','M']
for i, txt in enumerate(vert_labels):
#plt.annotate(txt, # this is the text
plt.annotate
(f'{txt}\n({tri_coords[0,i]:.0f},
{tri_coords[1,i]:.0f})',
(tri_coords[0,i], tri_coords[1,i]),
                # this is the point to label
 textcoords="offset␣points",
                # how to position the text
xytext=(0,10),
                # distance from text to points (x,y)
```

```
 ha='center')
                # horizontal alignment can be left, right or

# use set_position
ax = plt.gca()
ax.spines['top'].set_color('none')
ax.spines['left'].set_position('zero')
ax.spines['right'].set_color('none')
ax.spines['bottom'].set_position('zero')
#plt.xlabel('$x$')
#plt.ylabel('$y$')
plt.legend(loc='best')
plt.grid() # minor
plt.axis('equal')
plt.show()
```