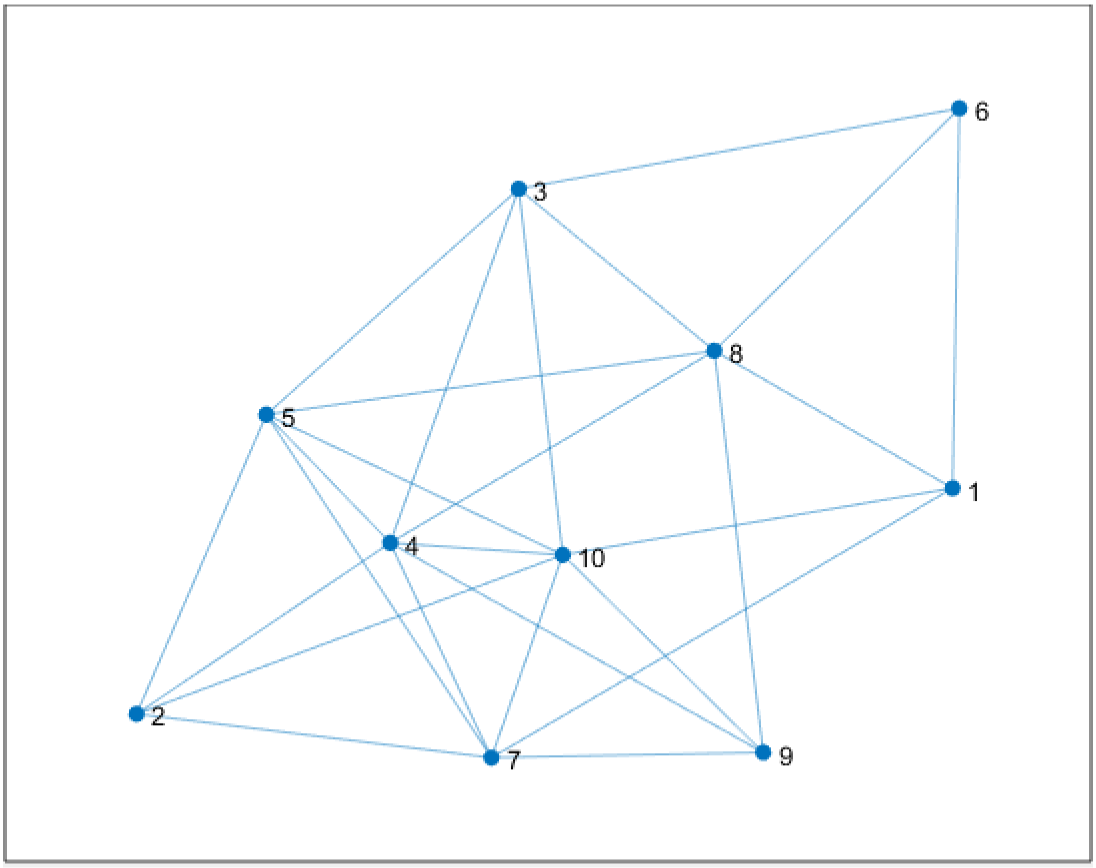
MS 4117 Project

Name: Cathaoir Agnew **Student ID:** 16171659

Q1 Plot the Graph



Q2 Is g Eulerian? Or does it have an Euler path? (check degrees)

- To be Eulerian, the degrees of all vertices must be even.

Since vertex 3 is of degree 5, this graph is not Eulerian

- To have an Euler path, it must have at most 2 vertices of odd degree

There are 4 vertices of odd degree, so this graph does not contain an Euler Path

The degrees of the vertices are in ascending order:

4 4 5 7 6 3 6 6 4 7

Q3 Is g bipartite? (check eigenvalues)

- A graph is bipartite if and only if the collection of eigenvalues is symmetric about zero

The eigenvalues of the graph are below:

-2.8629

-2.1141

-1.3833

-1.2136

-0.8126

-0.4066

0.4762

0.8931

1.9033

5.5204

It is clear that this is not the case. So the graph is not bipartite.

Q4 How many spanning trees has g? Plot one of them.

To get total number of spanning trees, I got the Laplacian matrix of the graph. I then removed the first row and the first column of the Laplacian matrix giving a sub-block of the Laplacian matrix. I then calculated the cofactor of this sub-block of the Laplacian matrix and then took the absolute value of the cofactor. The cofactor is calculated by:

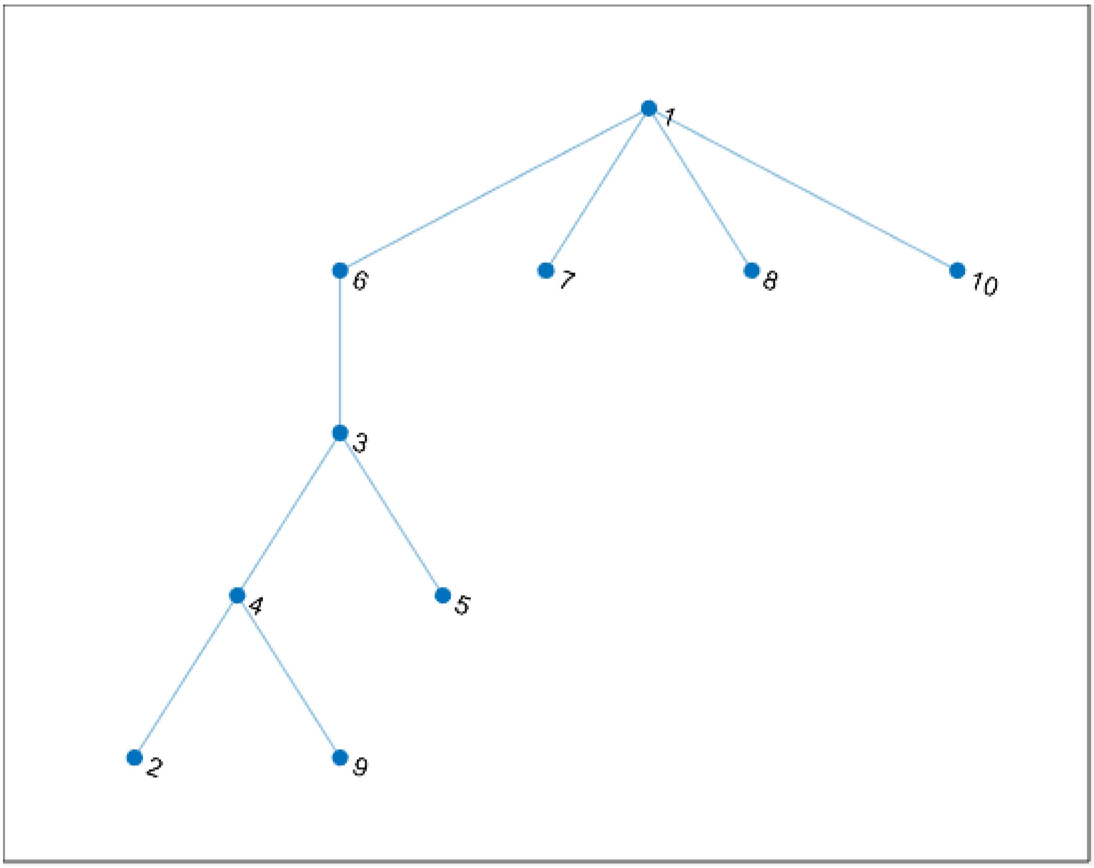
```
cofactorA = transpose(det_lap*inv_lap)
```

This gave me a result of: 1.0e+05

The total number of spanning trees is 100,000.

Below figure Figure 1, is one of the spanning trees.

Figure 1



Q5 Find the eigenvalues of the spanning tree you plotted. How does its maximum eigenvalue relate to the maximum eigenvalue of g?

To find the eigenvalues of the above graph of the spanning tree Figure 1, I used Matlab's built in function to give me the adjacency matrix. I then got the eigenvalues of this matrix.

The values of the eigenvalues of the spanning tree are below:

```
 -2.2047
  -1.8039
  -0.9408
       0
       0
       0
       0
   0.9408
   1.8039
   2.2047
```

The maximum eigenvalue for the spanning tree = 2.2047
The maximum eigenvalue for the original graph g = 5.5204

It is clear that the eigenvalue of the original graph g is greater than the eigenvalue of the plotted spanning tree. 5.5204 > 2.2047

Q6 Find upper and lower bounds for the chromatic number χ and the independence number α and find a proper minimal vertex colouring and maximal independent vertex set (you will need to do this by hand).

To find the lower bound of the chromatic number, χ , it is calculated by: $1 - \frac{\lambda_1}{\lambda_n}$

Where λ_1 is the largest eigenvalue of the original graph g and λ_n is the smallest eigenvalue of the original graph g.

Lower bound = $1 - \frac{5.5204}{-2.8629}$ = 2.928 ≈ 3

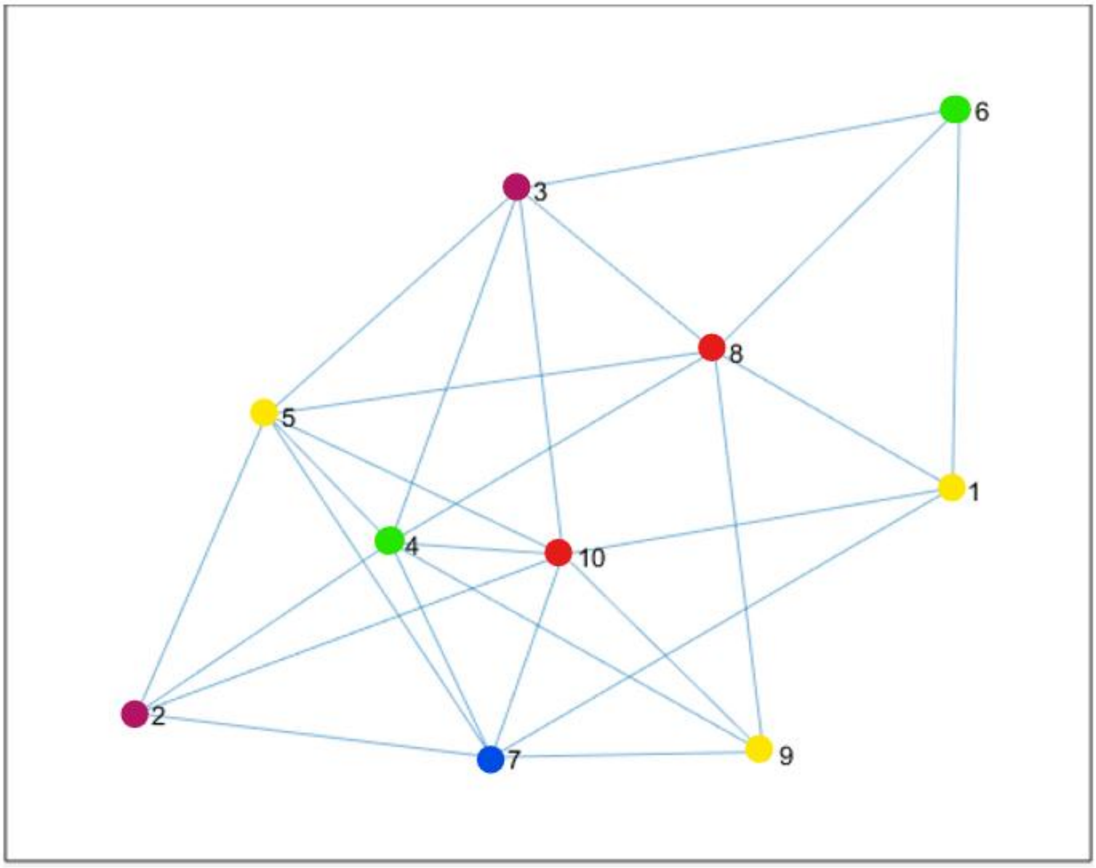
To find the upper bound of the chromatic number, χ , it is calculated by: $1 + \lambda_{Largest}$

Upper bound = 1 + 5.5204 = 6.5204 ≈ 7

So the chromatic number is within the interval

$$3 \leq \chi \leq 7$$

Using trial and error. I found it was $\chi = 5$



The upper bound for the maximum number of independent number of vertices, α is given

$$\alpha \leq \min \{ \mathrm{n} - n_+, n - n_- \}$$

- n- is the number of negative eigenvalues, from original graph
- n+ is the number of positive eigenvalues, from original graph
- n is the number of vertices in the graph

For the graph we have: n=10, n+ = 4, n- = 6 So α is the min of {6, 4}

Upper Bound α = 4

The lower bound of the maximum number of independent vertices is gotten by

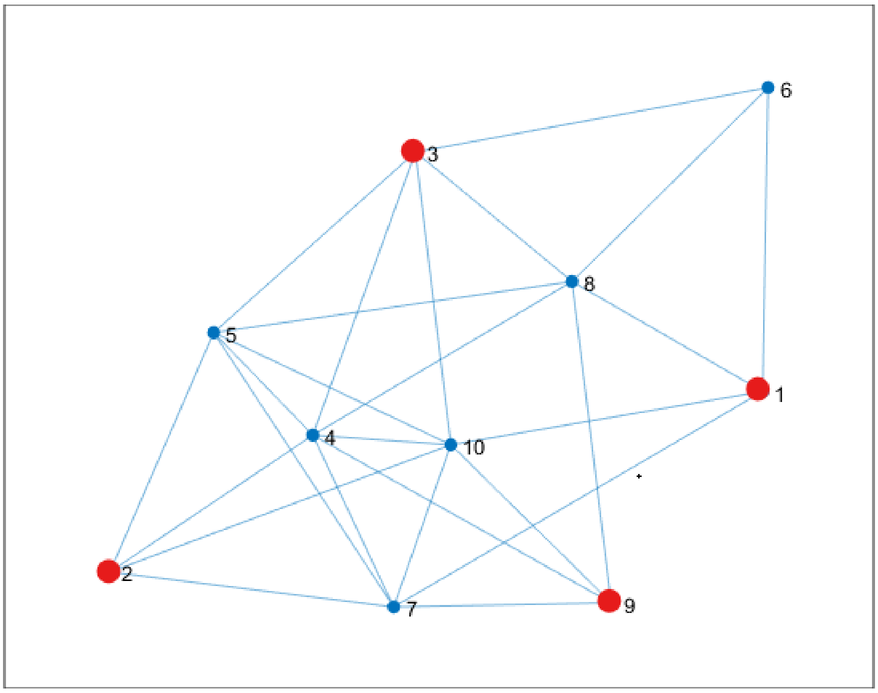
$$\alpha \geq \frac{n}{\chi}$$

So the lower bound $\alpha = \frac{10}{5}$ = 2

$$2 \leq \alpha \leq 4$$

From colouring in the graph below, we can see that I found 4 independent vertices. Since 4 is within my interval this must be the maximum independent vertices.

The maximum set of independent vertices I found was (1,2, 3, 9).



Q7 Find the algebraic connectivity

- The algebraic connectivity, μ , is the second smallest eigenvalue of the Laplacian

So μ = 2.1958

MATLAB CODE

```
rng(16171659,'twister'); %replace 12345678 by your student id
G = round(rand(10));G = triu(G) + triu(G,1)';G = G - diag(diag(G));
g=graph(G); % you can label if you wish

plot(g)  % plotting graph

%1 eulerian
 degrees = degree(g) % checking degrees

%2 bipartite, checking eigen values
eigen = eig(G)

%3
%laplacian matrix
lap = laplacian(g);

% used for connectivity in last part
eigenLap = eig(lap)


%sub matrix of laplacian matrix
%remove first row
lap(1,:)=[];
%remove first col
lap(:,1)=[];

% det & inverse calc
det_lap = det(lap);
inv_lap = inv(lap);

% cofactor
cofactorA = transpose(det_lap*inv_lap);
% abs value of cofactor
absCoFactorA = abs(cofactorA)

% graph one span tree
MinTree = minspantree(g);
plot(MinTree)

%getting adjaceny matrix of the spanning tree
AdjMinTree = adjacency(MinTree)
%Calculating eigen values of the adjaceny matrix
eigenAdjMinTree = eig(AdjMinTree)
```