MT 112, Project #2 (Counting Systems), Spring 2020   Name ___________________________________

- You will not be expected to present this project in class.
- Format for submission: You may use any of the following formats.
    - 5-page paper (counting images)
    - 15-20 slide PowerPoint (or other presentation program such as Prezi)
    - Poster (16x20 or larger)
        - Note that the poster format cannot be submitted electronically
    - If you have another creative suggestion, please ask!
- Due Date is April 7, 2020.

For this project you should select one type of graph theory problem to consider: Euler-type, Hamilton-type or tree/network type to analyze. Examples are shown at the end to illustrate the type of problem and level of complexity required, but you are not required to use these examples and may find your own problem of a similar type.

Your project should include information on (for all types):
- Any real-world scenarios should be converted to an abstract graph
- Each vertex of the graph should be analyzed for the degree of the vertex
- The overall type of graph should be characterized if it falls into any of the special types discussed (e.g. is it complete?)
- Identify a real-world application where your analysis would be useful (most graph types come with their own in the names, but you should articulate it in a paragraph or two, or provide additional applications)

Other elements of the project will depend on the type of graph/problem selected.

Euler-type Chinese Postman problems:
- Identify the location of your "post-office"; or the starting and stopping locations of your parade.
- Determine the number of odd vertices in your graph
- Calculate the minimum number of edges you would need to add to Eulerize (or semi-Eulerize) the graph
- Identify at least two different paths for your postman or parade
    - If you are doing a parade, you will also need to determine how many blocks long your parade is, and make sure that you can complete the circuit without running into the trailing end of the parade train.

Hamilton-type Traveling Salesman problems:
- If the graph is complete, calculate the number of unique circuits exist on the graph.
- If the information for the graph is in a table, you may work with the table, but you should create a graph that represents the same information; conversely, if you start with a graph, you should represent that graph in table format
- Apply each of our approximation Algorithms to the graph
    - Repeated Nearest Neighbor
    - Kruskal's Algorithm
- Analyze the number of steps taken in each algorithm for the size graph you have

- If your graph is not complete, you will need to determine if any Hamilton circuits exist, and how many, so while you can do this, I generally discourage it as it makes the problem more complex.
- Identify the lowest cost circuit you are able to find
- Calculate the number of circuits you would need to calculate to check your result with the Brute Force algorithm.

If you select a tree/network graph type:

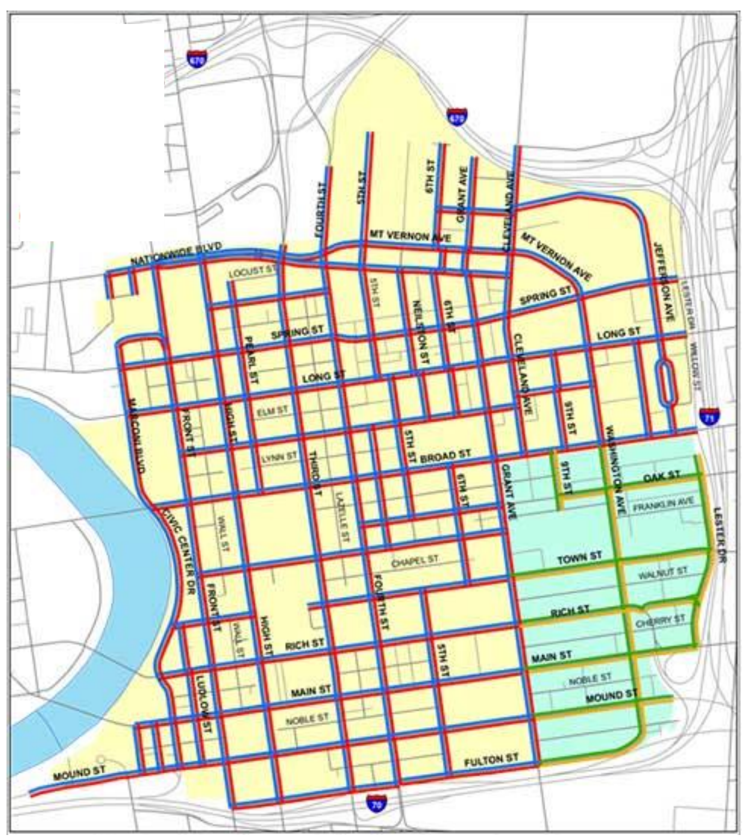
- If your graph is not already a tree, identify the number of vertices, and the number of edges required to make a spanning tree, and the redundancy of the graph
- Use Cheapest Link to find the minimum cost spanning tree
- Explain why the tree algorithm is not approximate
- You may use a map table initially, but as with the Hamilton graph type, you will need to convert your table to a graph in order to visualize the information, although you may use the table for analysis
- Summarize some of the modern applications of network theory

It is strongly suggested that you send me a paragraph (by email is fine) about your selected topic by March 31st, 2020 to make sure that your selection is viable, and to avoid any miscommunication.

You will be required to cite sources (citation format is up to you, but I recommend using the References features in Word since you just have to enter the info and it will format your bibliography automatically).

This project is worth 50 points.

Euler neighborhood graph example:



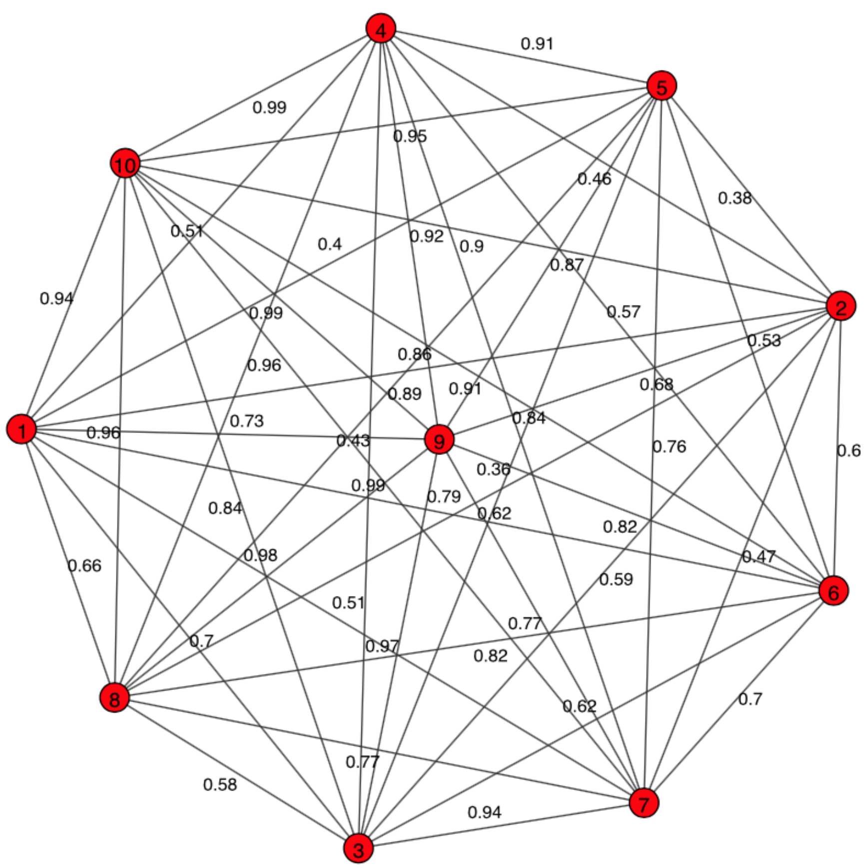
Hamilton Type:

Table version:

Distance Table (approximate kilometres)

	Beaufort West	Bloemfontein	Britstown	Capte Town	Colesberg	De Aar	Durban	East London	Gaborone	George	Graaff-Reinet	Grahamstown
Bloemfontein	544	.	398	1004	226	346	634	584	622	773	424	601
Cape Town	460	1004	710	.	778	762	1753	1079	1501	438	787	899
Colesberg	318	226	195	778	.	143	860	488	848	547	198	375
Durban	1178	634	1032	1753	860	980	.	674	979	1319	942	854
East London	605	584	609	1099	488	557	674	.	1206	645	395	180
Gaborone	1042	622	791	1501	848	843	979	1206	.	1361	1012	1223
George	258	773	509	438	547	571	1319	645	1361	.	349	465
Grahamstown	492	601	496	899	375	444	854	180	1223	465	282	.
Harare	2066	1522	1849	2526	1748	1868	1702	2106	1056	2295	1946	2123
Johannesburg	942	398	725	1405	624	744	578	982	358	1171	822	999
Kimberley	504	177	253	962	292	305	811	780	538	762	490	667
Ladysmith	954	410	808	1413	636	756	236	752	755	1183	834	932
Mafikeng	884	464	633	1343	672	685	821	1048	158	1203	854	1065
Maputo	1349	897	1289	1900	1123	1243	625	1301	957	1670	1321	1478
Maseru	609	157	555	1160	383	503	590	630	702	913	599	692
Mbabane	1129	677	1075	1680	903	1023	562	1238	719	1450	1101	1418
Port Elizabeth	501	677	572	769	451	520	984	310	1299	335	291	130
Pretoria	1000	456	783	1460	682	802	636	1040	350	1229	880	1057
Umtata	713	570	688	1314	517	636	439	235	1192	880	503	415
Welkom	697	153	551	1156	379	499	564	737	479	926	577	754
Windhoek	1629	1593	1379	1500	1573	1430	2227	1987	1735	1887	1697	1856

Graph version:



Tree/Network Graph: