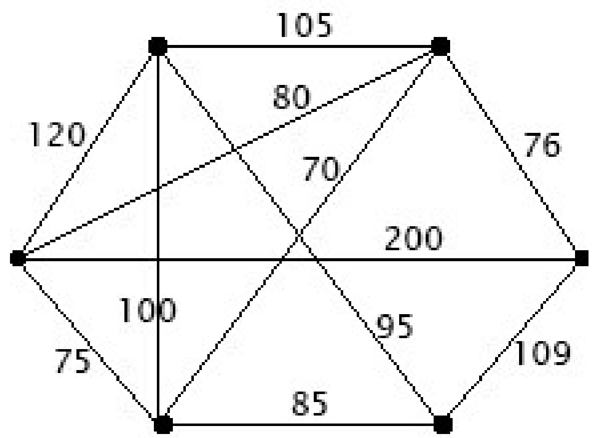
Math 13 – Liberal Arts Math HW 2 (due Thurs, 2/14) Name____________________________

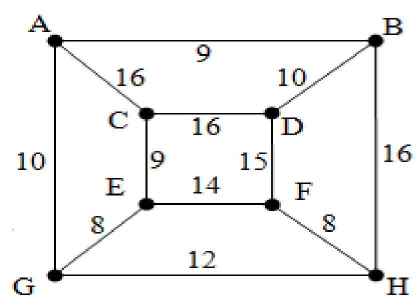
Write all responses on separate paper. Use complete sentences to explain your thinking. Show your work for credit.

1. Construct a complete graph on 6 vertices and use a squiggly line to indicate one possible Hamiltonian circuit for this graph. How many such Hamiltonian circuits are there?

2. Construct a minimal spanning tree of the graph given below




3. Consider the weighted-edge graph shown at right.




   a. Find the traveling salesman path starting from the upper left vertex and using the nearest neighbor algorithm.

   b. What happens when you try to find the traveling salesman path using the shortest edge algorithm? Explain.

   c. Use brute force to determine the shortest Hamiltonian path.

   d. Find a minimal spanning tree for this graph

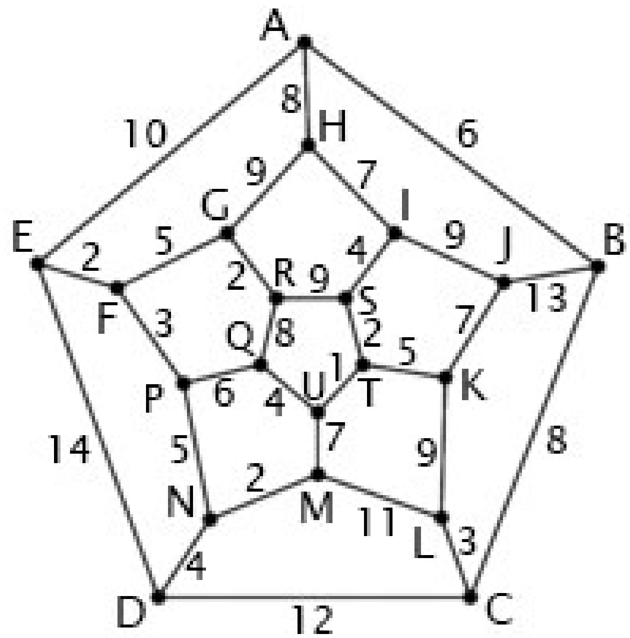
4. Consider the weighted-edge graph shown at right.

   a. Find the traveling salesman path starting from vertex A and using the nearest neighbor algorithm.

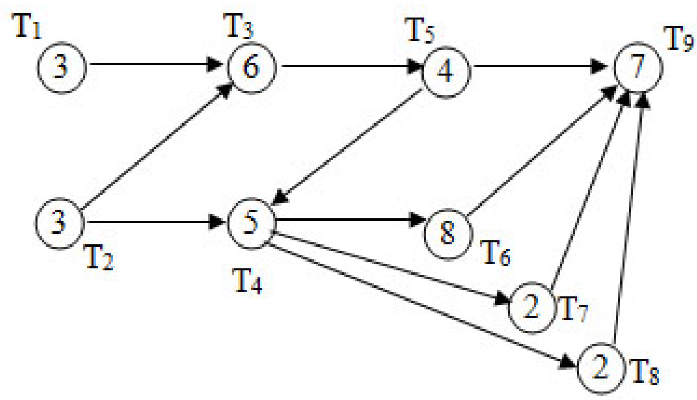
   b. Add additional edges, as needed, to make a complete graph for this graph of 20 vertices. Weight all the new edges at 100. Find the traveling salesman path using the shortest edge algorithm.

   c. Can you find a shorter Hamiltonian circuit?

   d. Find a minimal spanning tree for this graph

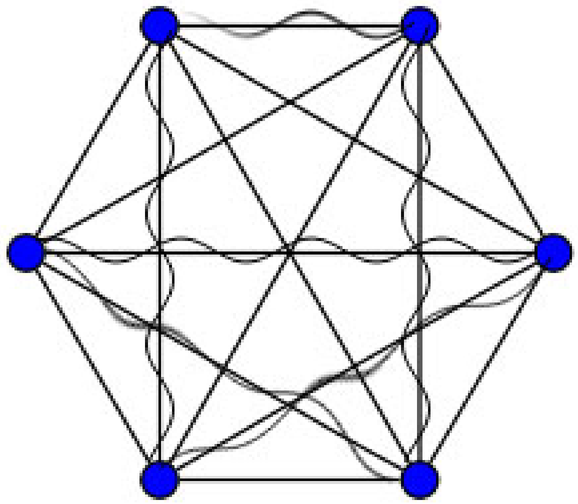



5. Given the order requirement digraph shown at right (with times in minutes).
   a. Follow the list processing algorithm to schedule the tasks to two processors the priority list $T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_8, T_9$
   b. What is the critical path for this digraph?
   c. Use the critical path scheduling to prioritize the tasks and then use this priority list to schedule the tasks for two processors.
   d. Is your schedule optimal? If not, how could you improve on it?

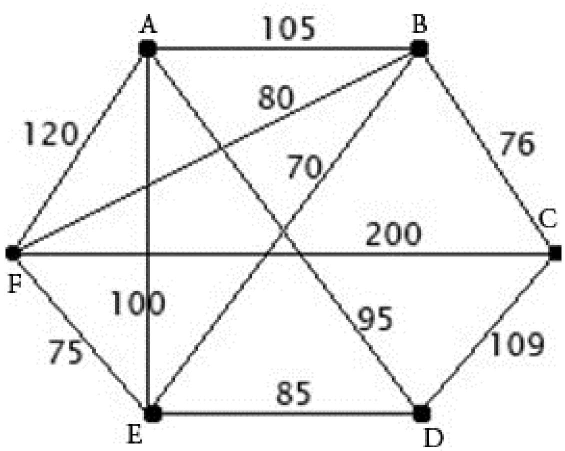


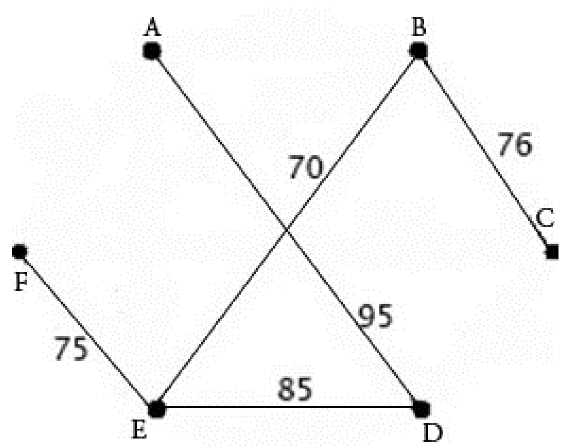
## Math 13 – Liberal Arts Math HW 2 Solutions.

1. Construct a complete graph on 6 vertices and use a squiggly line to indicate one possible Hamiltonian circuit for this graph. How many such Hamiltonian circuits are there?
   ANS: There are 5! = 120 different ways to make a Hamiltonian path on K6. Below is one of them.



2. Construct a minimal spanning tree of the graph given below

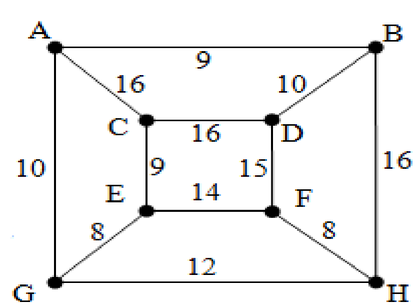



SOLN: It helps to label the vertices so you can refer to them. The sorted edges can be described as BE (70), FE (75), BC (76), BF (80), DE (85), DA (95), AE (100), AB (105), DC (109) FA (120) and FC (200). Applying the sorted edges algorithm, we arrive at BE (70), FE (75), BC (76), DE (85) and DA (95) as a minimal spanning tree.

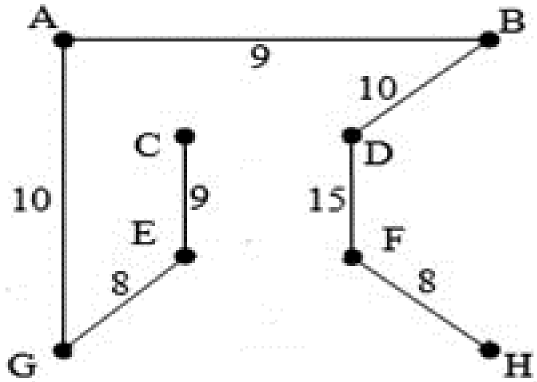
3. Consider the weighted-edge graph shown at right.

   a. Find the traveling salesman path starting from the upper left vertex and using the nearest neighbor algorithm.
   ANS: ABDFHGECA



b. What happens when you try to find the traveling salesman path using the shortest edge algorithm? Explain.

ANS: The sorted edges are GE, FH (8), AB, CE (9), AG, BD (10), GH (12), FE (14), DF (15) CD, BH and AC (16) . Adding these, in turn, from smallest to largest avoiding valences greater than 2 and circuits leads to a dead end:




Note that the algorithm is only guaranteed to produce a result if the graph is complete. This is not a complete graph.

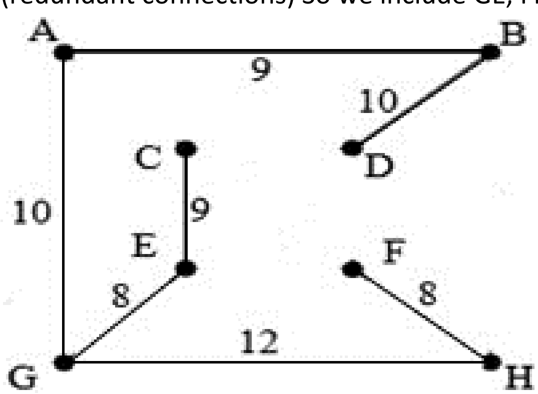
c. Use brute force to determine the shortest Hamiltonian path.

These are all the different Hamiltonian paths and their lengths:

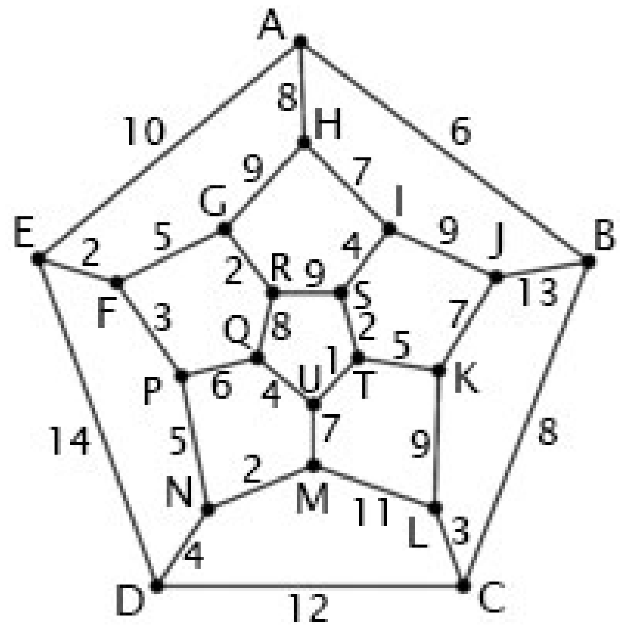
ABDCEFHGA = 88
ABHFDCEGA = 91
ABDFHGECA = 87
ACDBHFEGA = 98
ACEFDBHGA = 102
ACDFEGHBA = 106

d. Find a minimal spanning tree for this graph

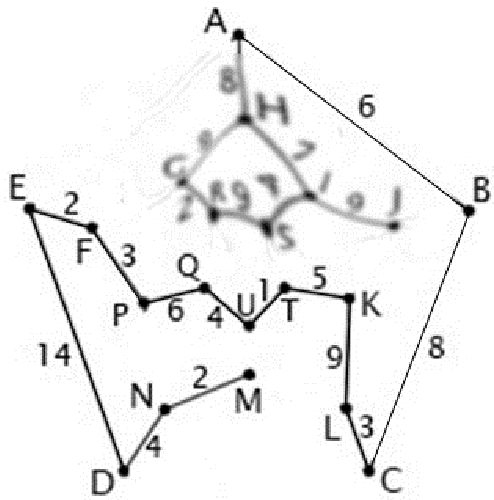
Again, the sorted edges are GE, FH (8), AB, CE (9), AG, BD (10), GH (12), FE (14), DF (15) CD, BH and AC (16) . Now we're not restricted to valences less than 3, but we must avoid circuits (redundant connections) So we include GE, FH



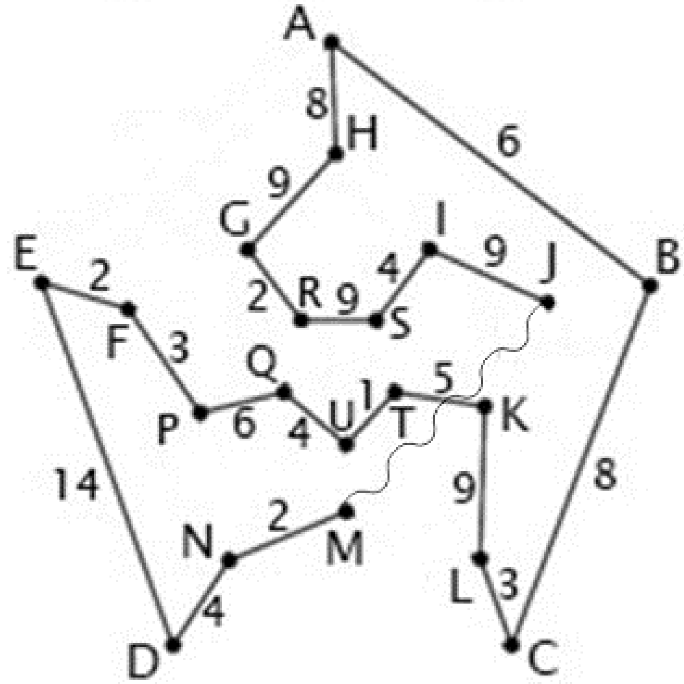
4. Consider the weighted-edge graph shown at right.




a. Find the traveling salesman path starting from vertex A and using the nearest neighbor algorithm.
   ANS: ABCLKTUQPFEDNM is as far as we get before becoming enisled at M. So we can't visit the cities labeled GHIJRS




b. Add additional edges, as needed, to make a complete graph for this graph of 20 vertices. Weight all the new edges at 100. Find the traveling salesman path using the shortest edge algorithm.
   Ans: It the nearest neighbor in the new graph will follow the same path as in part (a) to M where there is a choice of which 100 weight edge to follow. Do we go to H, G, R, S, I or J? Since J is the only vertex in the remaining subgraph with valence 1, that is a good place to start. Then following the nearest neighbor algorithm leads to the Hamiltonian circuit, ABCLKTUQPFEDNMJISRGHA, whose length is 209.




c. Can you find a shorter Hamiltonian circuit?
   ANS: One way to be certain is to do a brute force analysis. Ouch. Lets try finding patterns of Hamiltonian circuits here. One circuit is ABCDEFPNMLKJISTUQRGHA whose length is 128. That uses a heuristic of "exhaust the outer layers as much as possible while still leaving a way back out." Another heuristic that works is "Go in and out until you find yourself back home." This doesn't work so well. I'm interested in what other people find!

d. Find a minimal spanning tree for this graph

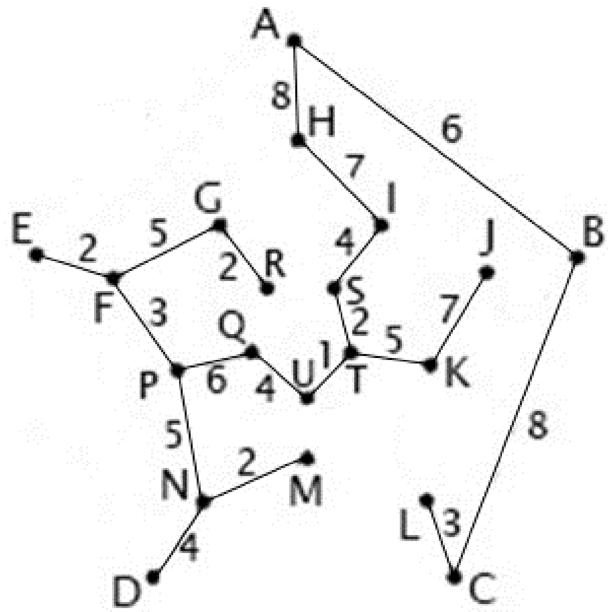
ANS: It may help to have a weighted incidence table.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | Q | R | S | T | U |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | | 6 | | | 10 | | | 8 | | | | | | | | | | | | |
| B | | | 8 | | | | | | | 13 | | | | | | | | | | |
| C | | | | 12 | | | | | | | | 3 | | | | | | | | |
| D | | | | | 14 | | | | | | | | | 4 | | | | | | |
| E | | | | | | 2 | | | | | | | | | | | | | | |
| F | | | | | | | 5 | | | | | | | | 3 | | | | | |
| G | | | | | | | | 9 | | | | | | | | | 2 | | | |
| H | | | | | | | | | 7 | | | | | | | | | | | |
| I | | | | | | | | | | 9 | | | | | | | | 4 | | |
| J | | | | | | | | | | | 7 | | | | | | | | | |
| K | | | | | | | | | | | | 9 | | | | | | | 5 | |
| L | | | | | | | | | | | | | 11 | | | | | | | |
| M | | | | | | | | | | | | | | 2 | | | | | | 7 |
| N | | | | | | | | | | | | | | | 5 | | | | | |
| P | | | | | | | | | | | | | | | | 6 | | | | |
| Q | | | | | | | | | | | | | | | | | 8 | | | 4 |
| R | | | | | | | | | | | | | | | | | | 9 | | |
| S | | | | | | | | | | | | | | | | | | | 2 | |
| T | | | | | | | | | | | | | | | | | | | | 1 |
| U | | | | | | | | | | | | | | | | | | | | |

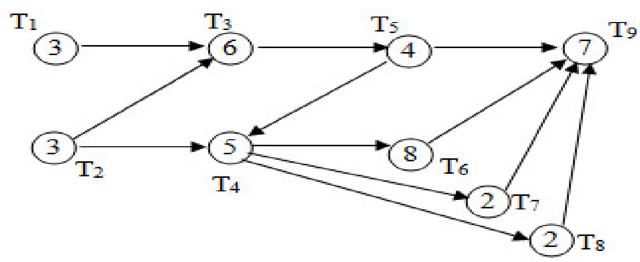
Ok, maybe that wasn't so terribly helpful. The total valence of the graph is 3*20 = 60 so the number of edges is 60/2 = 30 and we can tabulate the weights in increasing order like so:

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TU | 1 | | CL | 3 | | KT | 5 | | QR | 8 | | RS | 9 |
| ST | 2 | | DN | 4 | | AB | 6 | | AH | 8 | | AE | 10 |
| MN | 2 | | QU | 4 | | PQ | 6 | | BC | 8 | | LM | 11 |
| EF | 2 | | IS | 4 | | JK | 7 | | GH | 9 | | CD | 12 |
| GR | 2 | | NP | 5 | | MU | 7 | | KL | 9 | | BJ | 13 |
| FP | 3 | | FG | 5 | | HI | 7 | | IJ | 9 | | D# | 14 |

We add TU, ST, MN, EF, GR, FP, CL, DN, QU, IS, NP, FG, KT, AB, PQ, and JK without any circuits, but then avoid MU as that would create a circuit. Then we add HI, but avoid QR, adding AH and BC to complete the spanning tree as shown at right.




5. Given the order requirement digraph shown at right (with times in minutes).
   a. Follow the list processing algorithm to schedule the tasks to two processors the priority list $T_1$, $T_2$, $T_3$, $T_4$, $T_5$, $T_6$, $T_7$, $T_8$, $T_9$




a. SOLN:

| | 0–3 | 3–9 | 9–13 | 13–18 | 18–20 | 20–22 | 22–26 | 26–33 |
|---|---|---|---|---|---|---|---|---|
| P1 | T1 | T3 | T5 | T4 | T6 | | | T9 |
| P2 | T2 | | | | T7 | T8 | | |

b. What is the critical path for this digraph?
ANS: The critical path is $T_1T_3T_5T_4T_6T_9$, and has length 33.

c. Use the critical path scheduling to prioritize the tasks and then use this priority list to schedule the tasks for two processors.Is your schedule optimal? If not, how could you improve on it?
ANS The priority list obtained by critical path scheduling is T1T2T3T5T4T6T7T8T9. Note that the schedule achieved above in part (a) is optimal and can't be improved on, since it takes no more time than the critical path length.