Tutorial 9

Hamiltonian cycles

Homework

1. For *two* of the following graphs:
   - Q8 (1–5),
   - Q9,
   - Q10,

   determine a Hamiltonian cycle using each of the following methods:

   (i) Repetitive Nearest Neighbor,
   (ii) Cheapest Link,
   (iii) Nearest Insertion.
2. Solve *one* of the following questions: Q11 or Q12.

Question 1. Compute the following factorials

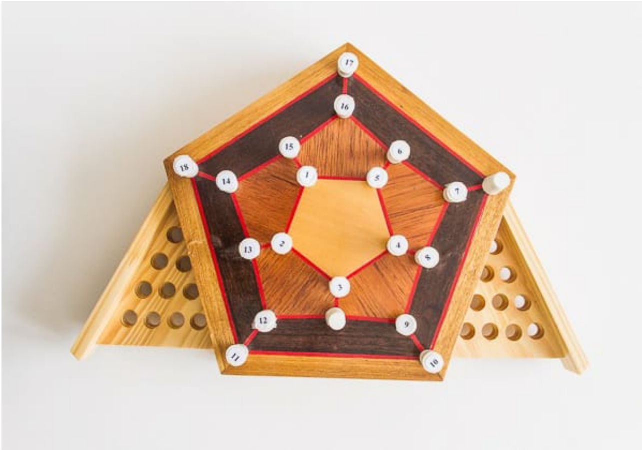
1. 8!
2. 12!
3. 16!

Question 2. Simplify the following factorials:

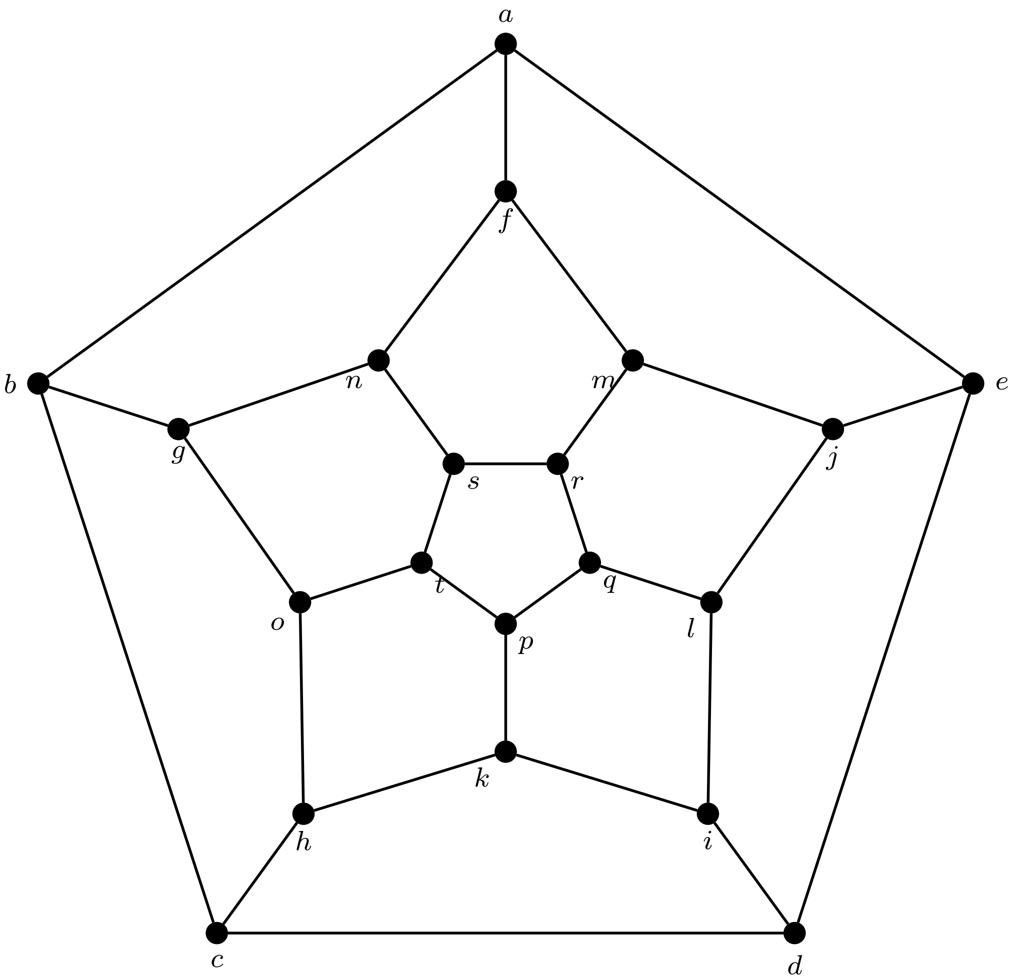
1. $9 \times 8!$
2. $\frac{11!}{8!}$
3. $6! \times \frac{7!}{5!}$

Question 3. How many different Hamiltonian cycles are there for K_4? K_8? K_{10}? Draw all possible Hamiltonian cycles for K_4.

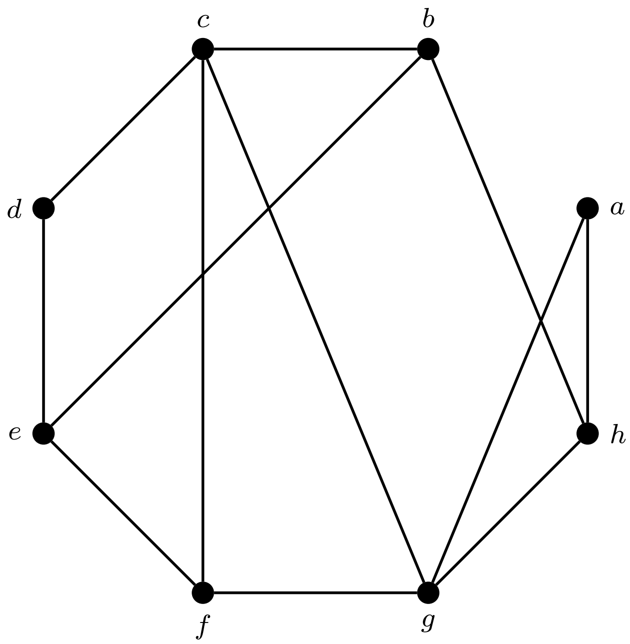
Question 4. Sir William Hamilton formalized the ideas of the Hamiltonian cycle and path. He posed this idea in 1856 in terms of a puzzle, which he later sold to a game dealer. The "Icosian Game" was a wooden puzzle with numbered ivory pegs where the player was tasked with inserting the pegs so that following them in order would traverse the entire board (as shown in the figure below).



This is equivalent to finding a Hamiltonian cycle in the following graph. Solve the problem.



Question 5. Find a Hamiltonian cycle for the following graph.



Question 6. For each of the graphs below, determine if G

(a) definitely has a Hamiltonian cycle;

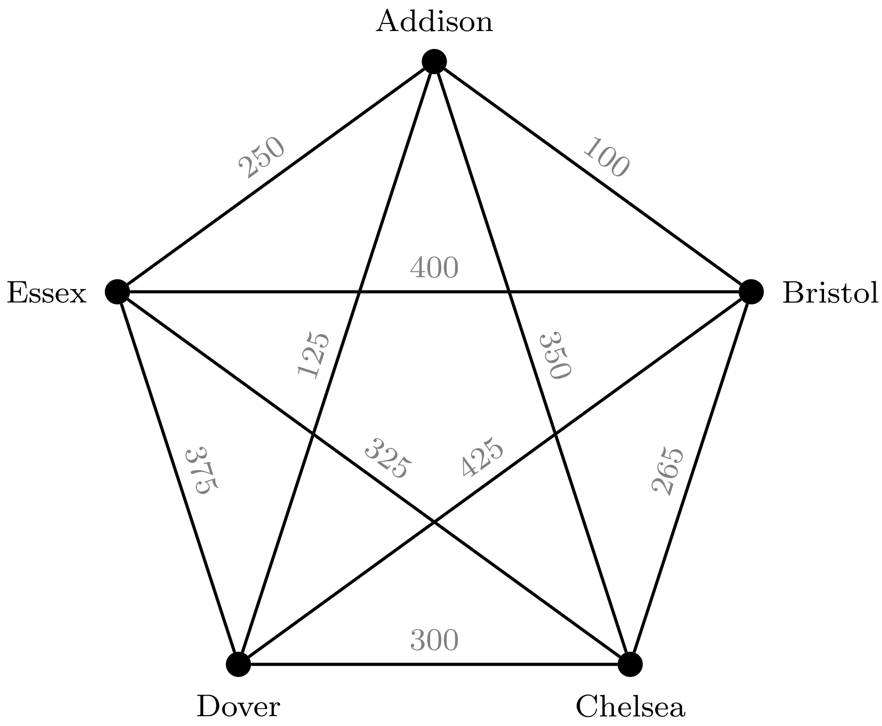
(b) definitely does not have a Hamiltonian cycle; or

(c) may or may not have a Hamiltonian cycle.

Explain your answer.

1. G has vertices of degree $3, 3, 3, 4, 4, 5$.
2. G is connected with 10 vertices, all of which have degree 6.
3. G has vertices of degree $1, 2, 2, 3, 5, 5$.
4. G is connected with vertices of degree $2, 2, 3, 3, 4, 4$.
5. G has vertices of degree $0, 2, 2, 4, 4, 5, 5$.

Question 7. Apply Repetitive Nearest Neighbor to the following graph.



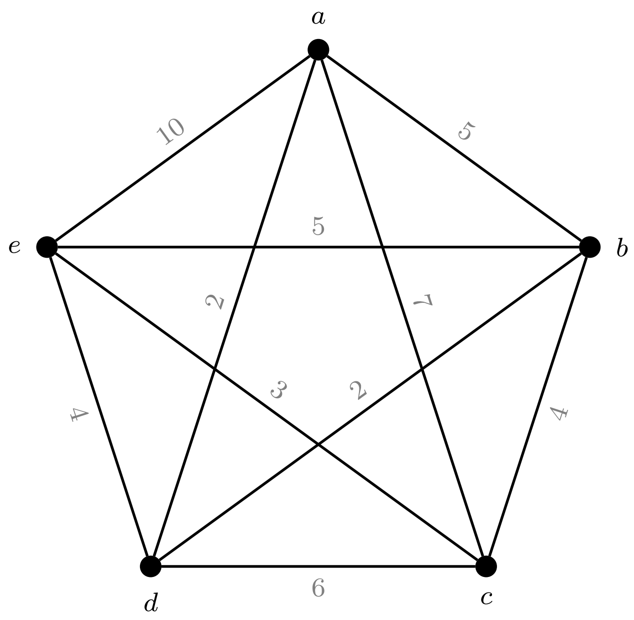
Question 8. Find a Hamiltonian cycle for each of the graphs below using:

(i) Repetitive Nearest Neighbor,

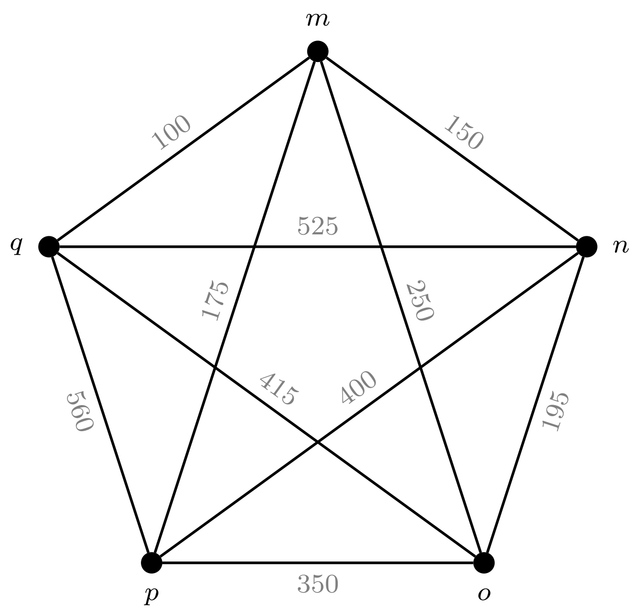
(ii) Cheapest Link, and

(iii) Nearest Insertion.

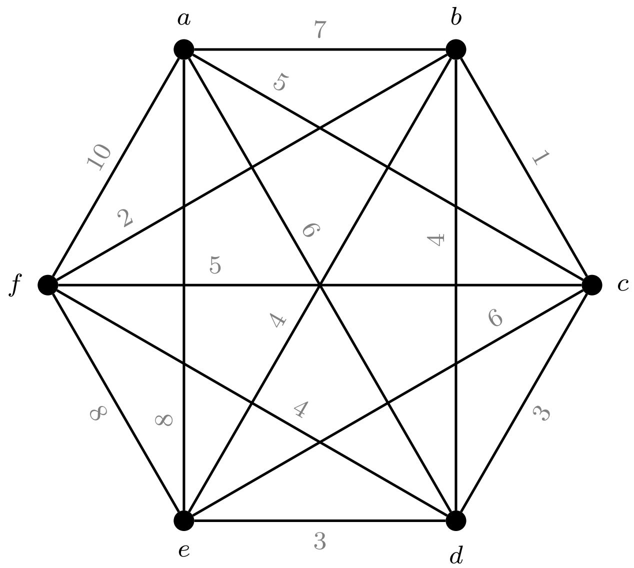
1.


a
b
c
d
e
10
5
5
2
7
3
2
4
4
6

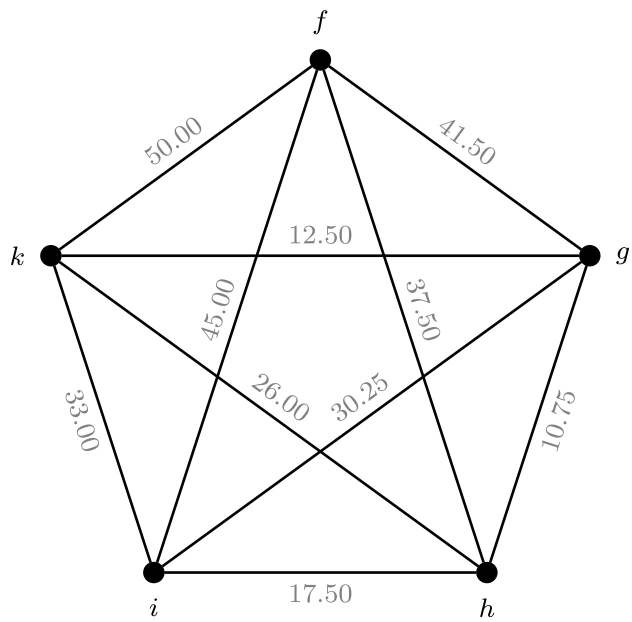
2.


m
n
o
p
q
100
150
525
175
250
415
400
560
195
350

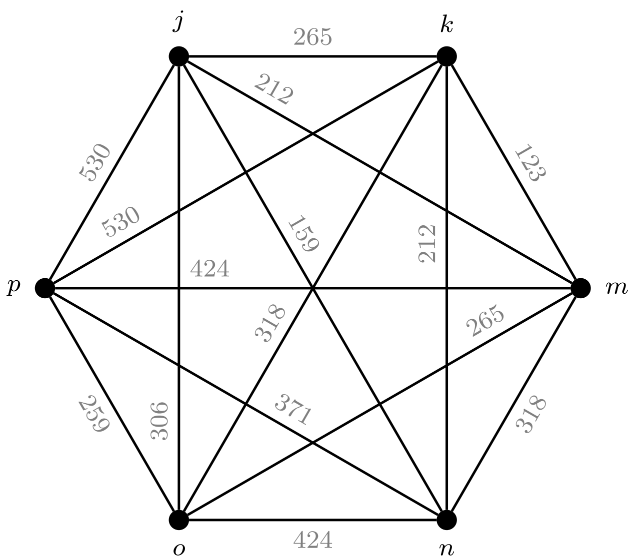
3.


a
b
c
d
e
f
7
5
10
1
2
6
4
5
4
6
8
8
4
3
3

4.


f
g
h
i
k
50.00
41.50
12.50
45.00
37.50
33.00
26.00
30.25
10.75
17.50

5.


j
k
m
n
o
p
265
212
530
123
530
159
212
424
318
265
259
306
371
318
424

Question 9. Chris wants to visit his 4 brothers over the holidays and has determined the costs as shown in the table below. Find a route (and its total weight) for Chris using

1. Repetitive Nearest Neighbor
2. Cheapest Link
3. Nearest Insertion

	Chris	David	Evan	Frank	George
Chris	.	325	300	125	100
David	325	.	215	375	225
Evan	300	215	.	400	275
Frank	125	375	400	.	305
George	100	225	275	305	.

Question 10. June and Tori are planning their annual winery tour of Virginia. They want to plan their route so they can see as many of the wineries in one day as possible and this year will be staying at the inn at Mt. Eagle Winery. The chart below lists the wineries and the time (in minutes) between each one. Find a possible route (and its total time) for June and Tori using

1. Repetitive Nearest Neighbor
2. Cheapest Link
3. Nearest Insertion

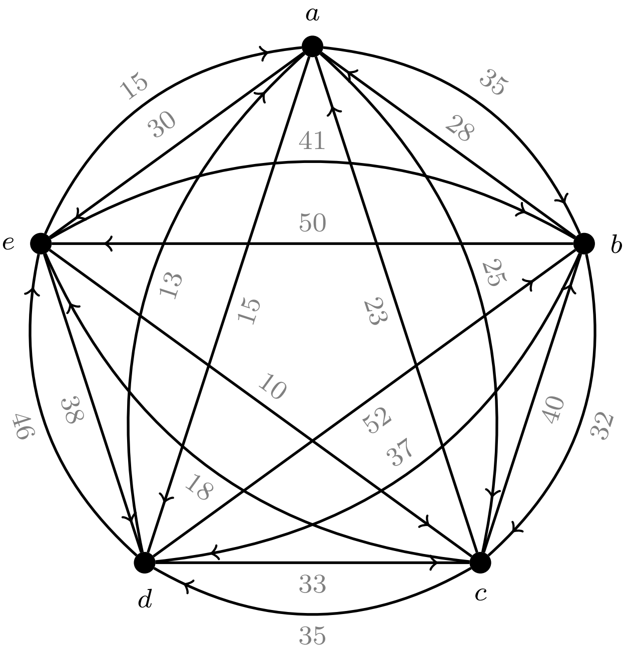
and determine if they can visit all six locations in one day.

	Bluebird Wines	Cardinal Winery	Elk Point Vineyard	Red Fox Wines	Graybird Vineyard	Mt. Eagle Winery
Bluebird	.	41	58	43	51	49
Cardinal	41	.	60	7	62	33
Elk Point	58	60	.	75	67	53
Red Fox	43	7	75	.	64	36
Graybird	51	62	67	64	.	68
Mt. Eagle	49	33	53	36	68	.

Question 11. Using the digraph below,

1. Apply the Undirecting Algorithm to find the weighted clone graph.
2. Using your result from 1, apply the Nearest Neighbor Algorithm with starting vertices a, a', c and c' and convert your results to directed cycles in the digraph. Find the total weight of each directed cycle.

3. Using your result from 1, apply the Cheapest Link Algorithm and convert your result to a directed cycle in the digraph and find its total weight.



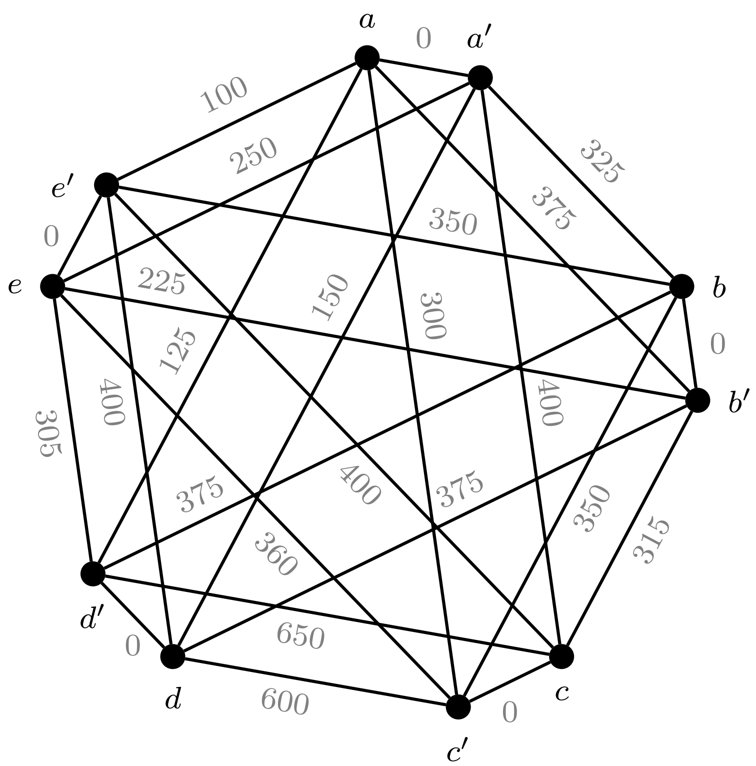
Question 12. Leena will be visiting her clients around Europe for the month of April. She has tried to estimate the cost of travel between two cities, using various modes of transportation and discovered the cost depends on the direction of travel. The table below gives these estimates.

1. Draw the directed graph representing the information in the chart below.
2. Apply the Undirecting Algorithm to find the weighted clone graph.
3. Using your result from 1, apply the Nearest Neighbor Algorithm with starting vertices a, a', d and d' and convert your results to directed cycles in the digraph. Find the total weight of each directed cycle.
4. Using your result from 1, apply the Cheapest Link Algorithm and convert your result to a directed cycle in the digraph and find its total weight.

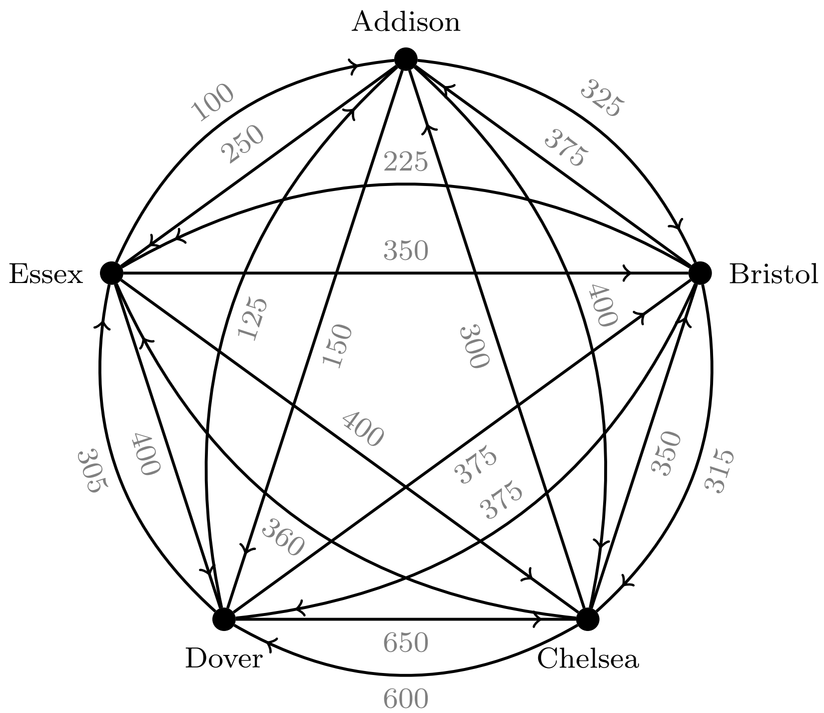
	Amsterdam	Bern	Düsseldorf	Genoa	Munich
Amsterdam	.	415	375	280	300
Bern	500	.	425	110	250
Düsseldorf	300	425	.	375	240
Genoa	150	200	500	.	400
Munich	275	350	315	400	.

Question 13. Explain why no cycles of length three exist in the graph resulting from applying the Undirecting Algorithm to a complete digraph.

Question 14. Determine a modification of Nearest Insertion that will allow it to be used on a graph obtained from a complete digraph using the Undirecting Algorithm. (Hint: the initial cycle should start from the lowest nonzero edge and should have length 4.) Use your modification on the weighted clone graph



corresponding to the following directed graph from lecture



Question 15. The Nearest Insertion Algorithm finds a Hamiltonian cycle by expanding smaller cycles through the addition of the closest vertex to that cycle. It suffers from the same problem as the other algorithms in that a large edge may be chosen in the last step of the algorithm. A

variation, called *Farthest Insertion*, first considers the vertices farthest apart since any Hamiltonian cycle must include both of them. In doing so, later additions of vertices will either reduce the cycle weight or increase it by small margins. The description of the algorithm appears below.

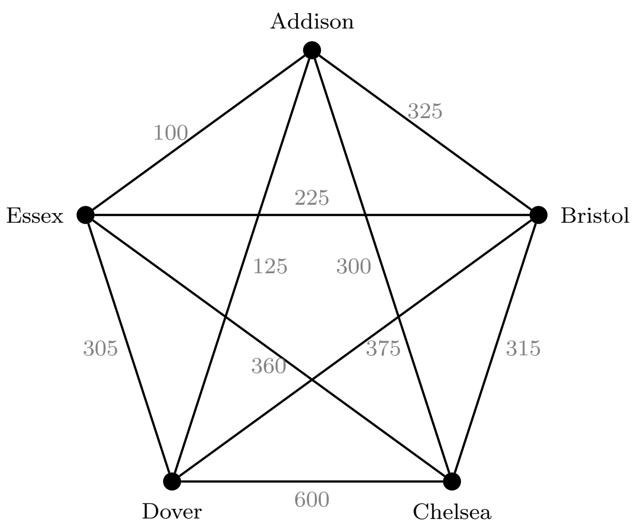
Farthest Insertion Algorithm
Input: Weighted complete graph $G = (V, E)$.
Steps:

1. Pick a starting vertex v_1.
2. Choose the vertex v_2 that has the highest weighted edge to v_1.
3. Form a list $(w_1, w_2, w_3, \ldots, w_n)$ where the entry in location i is the minimum weighted edge from v_i to either of v_1 and v_2. The entries for v_1 and v_2 will be left blank (denoted by $-$).
4. Choose vertex x with the largest value from the list created in Step 3. Form the cycle $v_1\, v_2\, x\, v_1$.
5. Update the list from Step 3 so the entries are now the weights from the chosen to unchosen vertices. Choose the next vertex y with largest value in the list.
6. Append the cycle of chosen vertices with y by removing one of the edges from that cycle. Determine which edges to add and subtract by choosing the lowest total as in Nearest Insertion; that is, if the cycle obtained from Step 4 was $a - b - c - a$ and d is the new vertex to add along with edge to dc, we calculate:
$$w(dc) + w(db) - w(cb) \quad \text{and} \quad w(dc) + w(da) - w(ca)$$
and choose the option that produces the smaller total.
7. Repeat Steps (5) and (6) until all vertices have been included in the cycle.

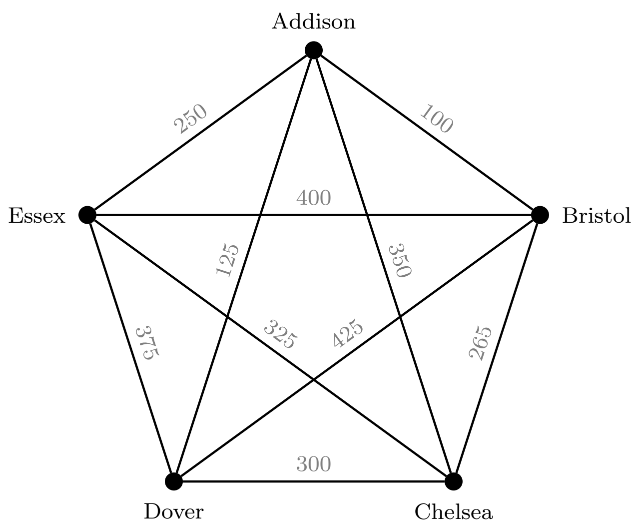
Output: Hamiltonian cycle.

Apply the Farthest Insertion to the following graph from the Lecture and from Question 7.

1.



2.



Question 16. Each morning a collection of online orders arrives at the warehouse of a large retailer. Steve, the warehouse manager, must ensure the items are packed and put onto the truck for shipment. However, the items are different every morning and are located in varying locations in the large warehouse. Steve has come to you for help in determining the best method for pulling stock from the shelves. Write a report detailing the Traveling Salesman Problem and how it applies to the warehouse. As part of your report, determine a route for the items shown in the map below. The route must start and end at the packaging bay (p) and the time required for moving down a long aisle is 45 seconds and down a short aisle or between aisles is 10 seconds. For example, it takes 85 seconds to get from item a to item b since four short segments and one long segment are used. Include a weighted graph and discussion of which algorithm(s) you used.