7.7 Exercise: In-class exercises are meant to introduce you to a new topic and provide some practice with the new topic. Work in a team of up to 4 people to complete this exercise. You can work simultaneously on the problems, or work separate and then check your answers with each other. You can take the exercise home, score will be based on the in-class quiz the following class period. **Work out problems on your own paper** - this document just has examples and questions.

7.7 Excursion: Hamiltonian Cycles & the TSP

7.7.1 Hamiltonian Cycles

Review & Hamiltonian cycle

Walk: A series of alternating nodes and edges traversing between adjacent nodes.

Path: A walk with no repeated vertices.

Circuit: A closed trail.

Trail: A walk with no repeated edges.

Cycle: A nontrivial circuit where the only repeated node is the first/last one.

Hamiltonial cycle: A Hamiltonian cycle in G is a cycle that uses every node of G. The graph G is called Hamiltonian if it contains a Hamiltonian cycle.

Eulerian trail or circuit: A trail or circuit where every edge is traversed [a]

[a] Discrete Mathematics, Ensley and Crawley

Question 1

For the following graph, trace out a Hamiltonian cycle. You can use the Hamiltonian Cycle visualization from
https://rachels-courses.github.io/Visualizations/ to practice.

Hamiltonian puzzle and Hamiltonial path problem

The Hamiltonian puzzle involves finding a Hamiltonian cycle in the edge graph of a dodecahedron.

Determining whether Hamiltonian cycles exist in graphs is the Hamiltonian path problem, which is NP-complete. [a]

NP complete means "nondeterministic polynomial time", where a solution to an NP-complete problem can be verified quickly, but there is no known efficient way to locate a solution in the first place. [b]

[a] From https://en.wikipedia.org/wiki/Hamiltonian_cycle

[b] From https://en.wikipedia.org/wiki/NP-complete_problem

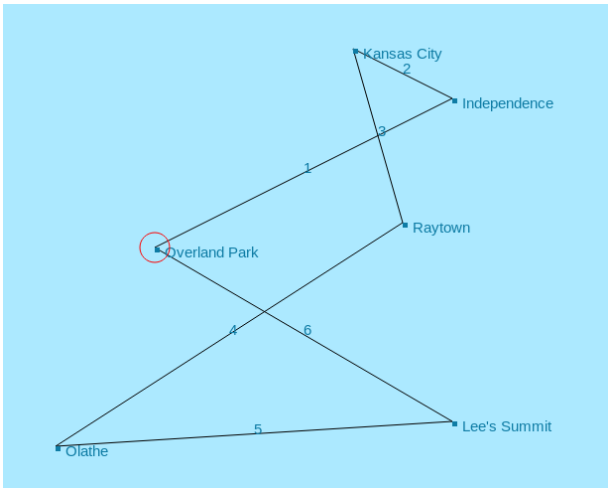
7.7.2 Travelling Salesperson

The "TSP" is another type of problem that asks the following: [a]

"Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city?"

[a] From https://en.wikipedia.org/wiki/Travelling_salesperson_problem

Go to https://rachels-courses.github.io/Visualizations/ and use the **Travelling Salesperson** visualization to help you work on this part.



Click on any node to begin the path. Click on subsequent nodes to add an edge between the nodes. The program records the distance (in pixels) at the bottom of the screen.

Question 2

If we have 6 total cities to visit, with the last city being visited twice, how many possible ways can we visit all cities? *(Hint: It's a counting problem...)*

1 (start)	5	4	3	2	1	1 (end)

Question 3

Come up with three different **cycles**, where all nodes are visited exactly once - except the starting point, which is visited first and last. Record the distances found for each. The starting city should be Overland Park.

#	Walk	Total distance
1.	OP → Olathe → Lee's Summit → Raytown → Independence → Kansas City → OP	1360

How did you go about trying to find a shortest path?

Question 4
Fill out the following table to record the distance between any two cities, given by the visualization.

	Indep.	K.C.	L.S.	Olathe	O.P.	Raytown
Indep.	0					
K.C.		0				
L.S.			0			
Olathe				0		
O.P.					0	
Raytown						0

Question 5
Using the table, generate another two paths as before.

#	Walk	Total distance
1.	OP → Olathe → Lee's Summit → Raytown → Independence → Kansas City → Overland Park	1360

What kind of method (if any) did you use to try to find a shortest path?