

Section 1.3 Hamiltonian Circuits and paths - continued

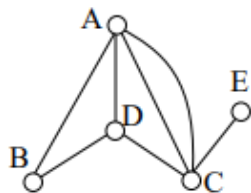
Recall:

What is a Hamiltonian circuit?

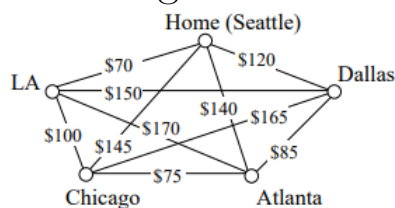
What is a Hamiltonian path?

Question:

Does a Hamiltonian path or circuit exist on the graph below?



With Hamiltonian circuits, our focus will not be on existence, but on the question of optimization; given a graph where the edges have weights, can we find the **optimal Hamiltonian circuit**; the one with lowest total weight.



We will learn 3 algorithms that help us find the optimal Hamiltonian circuit in a graph.

1 Brute Force Algorithm: (Exhaustive Search)

- Step 1: Make a list of **all Hamilton Circuits** from one starting vertex
- Step 2: Calculate the weight of each circuit.
- Step 3: Choose an optimal Circuit.

For complete graphs listing all Hamiltonian circuits can be very time consuming.

For K_4 you have to check _____ circuits.

For K_5 you have to check _____ circuits.

For K_6 you have to check _____ circuits.

Is this an efficient algorithm?

Brute Force is the only algorithm known at this time to always give an optimal circuit.

Example 1. You want to run errands around town. You want to start and end your errands at home and visit Target, the bank, and Starbucks, but not necessarily in that order. In the chart given below, the numbers represent miles between each destination.

	Home	Target	Bank	Starbucks
Home	*	5	3	1
Target	5	*	2	4
Bank	3	2	*	6
Starbucks	1	4	6	*

Draw a weighted graph from the information given in the chart.

List all of the possible Hamilton circuits and their weights.

What is the optimal circuit?

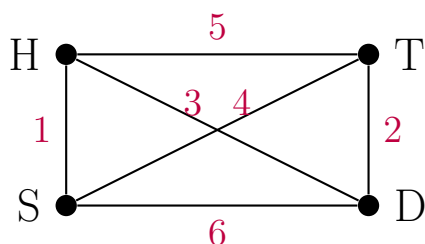
Approximate Algorithms:

2 Nearest Neighbor Algorithm (NNA)

- Step 1: Pick a starting vertex
- Step 2: When you arrive at a vertex, go to the nearest neighbor (go along the edge with the lowest weight) that has yet to be visited. Repeat to visit all vertices.
- Step 3: From the last vertex, return to the start.
- Step 4: Compute the weight of the circuit.
- Step 5: The circuit we obtain is called the **Nearest-Neighbor Circuit**.

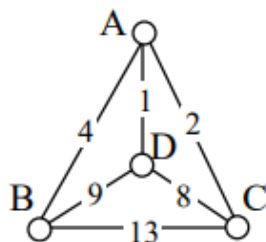
Example 2. (again): You want to run errands around town. You want to start and end your errands at home and visit Target, the bank, and Starbucks, but not necessarily in that order. In the chart given below, the numbers represent miles between each destination.

	Home	Target	Bank	Starbucks
Home	*	5	3	1
Target	5	*	2	4
Bank	3	2	*	6
Starbucks	1	4	6	*



Find the nearest-neighbor circuit using the graph.

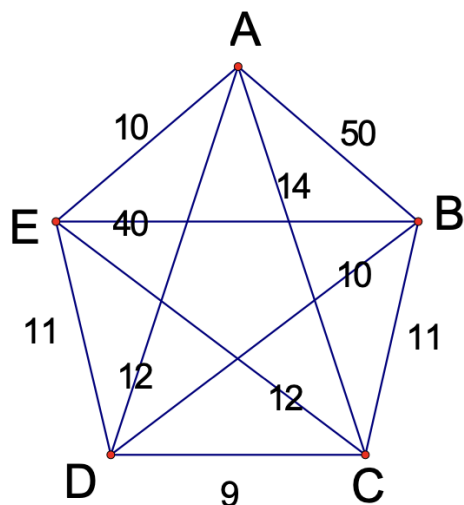
For this example NNA was successful! The NNC is optimal! **Example 3.** Apply the NNA algorithm to find a Hamiltonian circuit on the graph below.



For this example NNA was NOT successful! The NNC is NOT optimal!

Practice:

Example 4. A delivery service must deliver packages at Buckman (B), Chatfield (C), Dayton (D), and Evansville (E), and then return to Arlington (A), the home base. The following graph shows the estimated travel times (in minutes) between the cities.



1. Find the nearest-neighbor circuit for starting vertex A.
2. Find the nearest-neighbor circuit for starting vertex B.
3. Find the nearest-neighbor circuit for starting vertex C.
4. Find the nearest-neighbor circuit for starting vertex D.
5. Find the nearest-neighbor circuit for starting vertex E.

6. Which circuit is the best of all of these? Rewrite it with starting vertex A.

Repeated Nearest Neighbour Algorithm

What you just did for the previous example is called the Repeated Neighbour Algorithm.

Steps for RNNA:

1. Do the Nearest Neighbour Algorithm starting at each vertex.
2. Choose the circuit produced with minimal weight.