Section 1.3 Hamiltonian Circuits and paths - continued

Recall:

What is a Hamiltonian circuit?

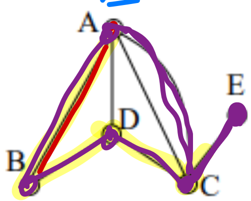
- a circuit that visits every vertex exactly once

What is a Hamiltonian path?

- a path that visits every vertex exactly once

Question:

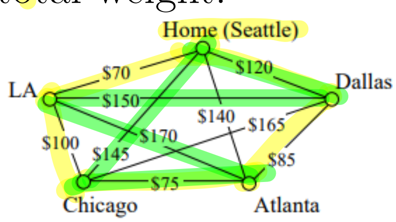
Does a Hamiltonian path or circuit exist on the graph below?



HP: ECDBA or ABDCE

no HC!

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.



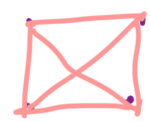
We will learn 3 algorithms that help us find the optimal Hamiltonian circuit in a graph.

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.

N vertices it had $(N-1)!$ HCs.

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



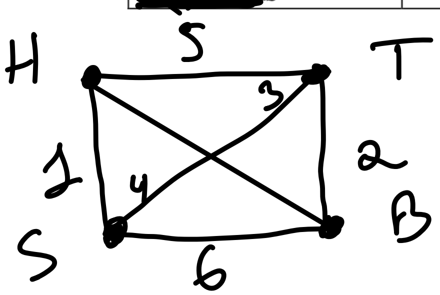
For K_4 you have to check $(4-1)! = 3! = 6$ circuits.
For K_5 you have to check $4! = 24$ circuits.
For K_6 you have to check $5! = 120$ circuits.

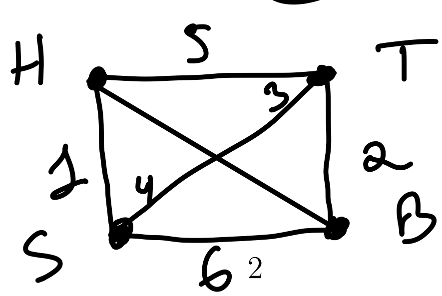
Is this an efficient algorithm? NO!!!

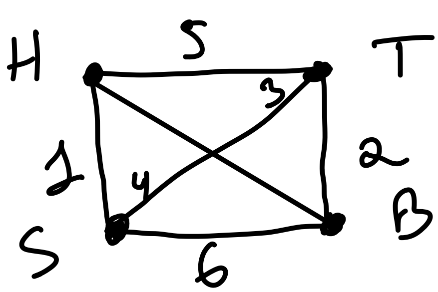
Brute Force is the only algorithm known at this time to always give an optimal circuit.

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.

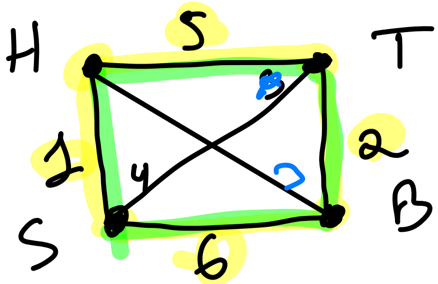
	Home	Target	Bank	Starbucks
Home	*	5	3	1
Target	5	*	2	4
Bank	3	2	*	6
Starbucks	1	4	6	*

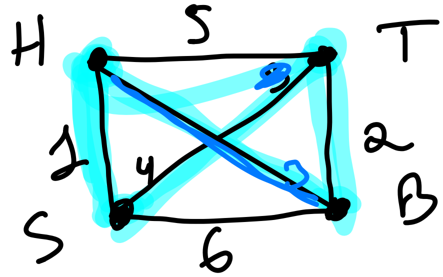


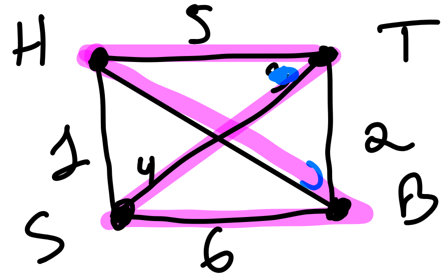




Draw a weighted graph from the information given in the chart.







6 HCs

List all of the possible Hamilton circuits and their weights.

HTBSH (or HSBTH its mirror circuit): 5 + 2 + 6 + 1 = 14

HBTSH (or HSTBH): 1 + 4 + 2 + 3 = 10

HTSBH (HBSTH) : 5 + 4 + 6 + 3 = 18

What is the optimal circuit?

HBTSH is the optimal circuit (or HSTBH)

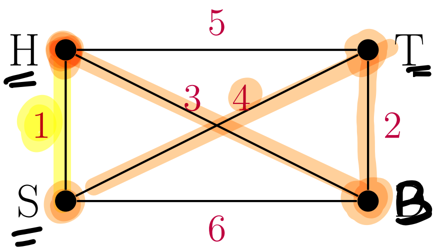
Approximate Algorithms:

2 Nearest Neighbor Algorithm (NNA)

- Step 1: Pick a starting vertex
- Step 2: When you arrive at a vertex, go ot 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	*



HSTBH NNC

1 4 2 3

weight: $1+4+2+3=10$

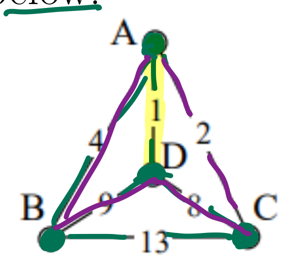
optimal circuit

Find the nearest-neighbor circuit using the graph.

HSTBH

for a weight of 10

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.



ADCBA → NNC

1 8 13 4

its weight is 26.

But circuit ACDBA

2 8 9 4

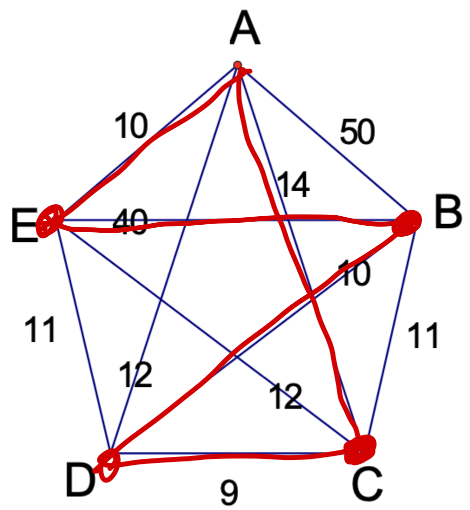
its weight is 23!

optimal one!

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.

A E D C B A is the NNC w/ weight 91 minutes
10 11 9 11 50

2. Find the nearest-neighbor circuit for starting vertex B.

B D C E A B is the NNC w/ 91 minutes.
10 9 12 10 50

3. Find the nearest-neighbor circuit for starting vertex C.

CDBEAC is the NNC w/ 83 minute
9 + 10 + 40 + 10 + 14

4. Find the nearest-neighbor circuit for starting vertex D.

DCBEAD
9 + 11 + 40 + 10 + 12 = 82 minutes

5. Find the nearest-neighbor circuit for starting vertex E.

EADCDE
10 + 12 + 9 + 11 + 40 = 82 minutes

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

Optimal circuit is starting at vertex D or E.

D C B E A D

A D C B E A

Repeated Nearest Neighbour Algorithm (RNNA)

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.