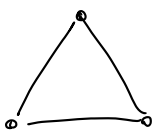
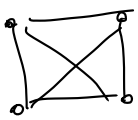
The Traveling Salesman Problem

is all about Ham. circuits in complete graphs.

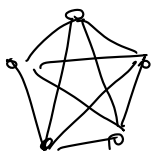
Complete graphs:



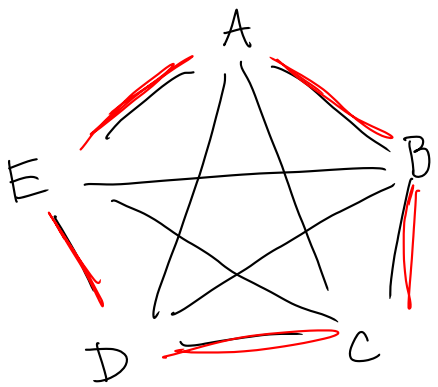
K_3



K_4



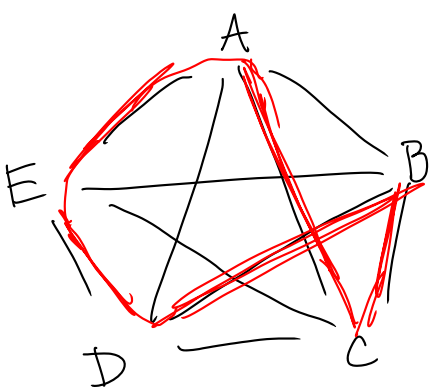
K_5



A Ham. circuit:

ABCDEA

OR:



ACBDEA

any permutation of these makes a Ham. circ

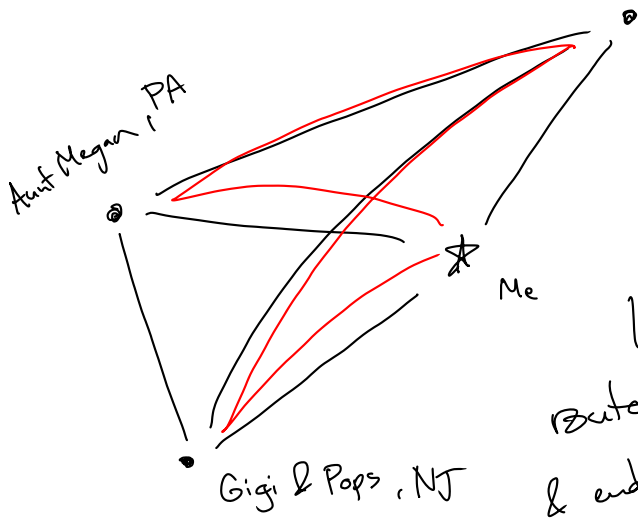
A D B C E A

A C B E D A

any permutation of B C D E gives another Ham. circuit

There are $4! = 24$.

Say I'm visiting 3 family members



What's the best route to visit everyone & end up back home?

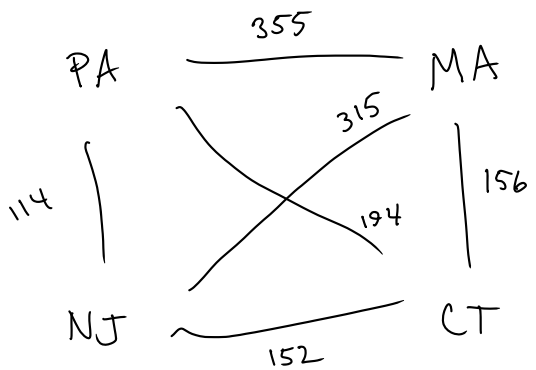
This is the Traveling Salesman Problem

Like the Amazon Delivery truck:

Every day, decide a new route to deliver my packages fastest. (with hundreds of addresses)

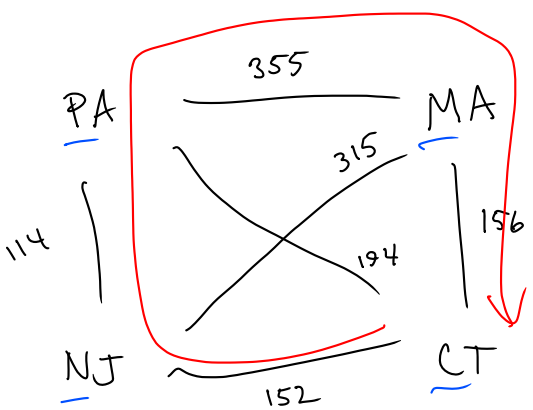
This is a TSP with ~200 vertices.
in K_{200}

A TSP is on a weighted graph
each edge has a weight = a number.



Find the Ham. circ. with least weight.

Simplest way to find the minimal Ham circ is to write all possible Ham. circs, & choose the best.

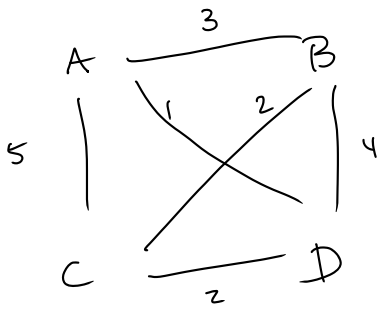


Hams	weight
CMNPC	156+315+114+194 = 779
CMPNC	156+355+114+152 = 777
CNMPC	152+315+355+194 = 1016
CNPMC	152+114+355+156 = 777
CPMNC	1016
CPNMC	779

Best route is: CMPNC
or CNPMC

This is called the brute-force algorithm

Find the best Ham. circuit:



Hams	weight
ABCDA	3+2+2+1 = 8
ABDCA	3+4+2+5 = 14
ACBDA	5+2+4+1 = 12
~~ACDBA~~	
~~ADBCA~~	
~~ADCBA~~	

ABCDA or ADCBA are the best!

Brute force takes a long time, even for computers! (with many verts)

If we have 200 verts, this makes 199! possible Hams to check.

↑ way too big to check all of them

Brute force works in theory, but is not practical for many verts.

No practical method is known

(also not known to be impossible)

IRL, we use "approximate" solutions.

Tricks for finding pretty-good routes.

(maybe not best)