Programming, Problem Solving, and Algorithms

CPSC203, 2019 W1

Announcements

Project 3 released soon. Due 11:59p, Nov 29.

“Problem of the Day” continues!

Today:

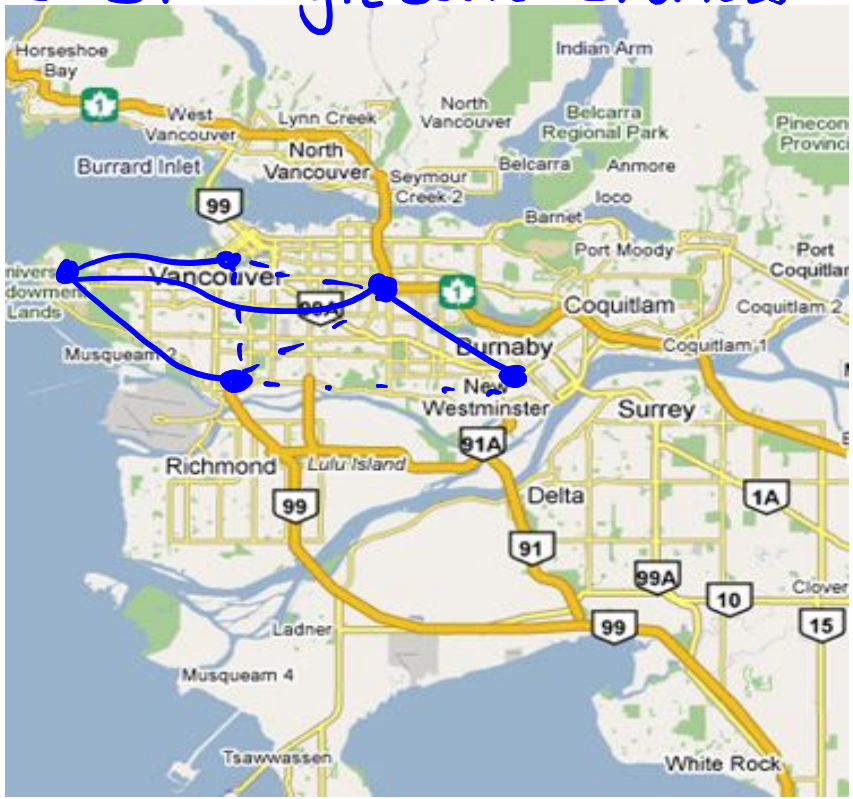
~~Shortest Path~~

Maps!

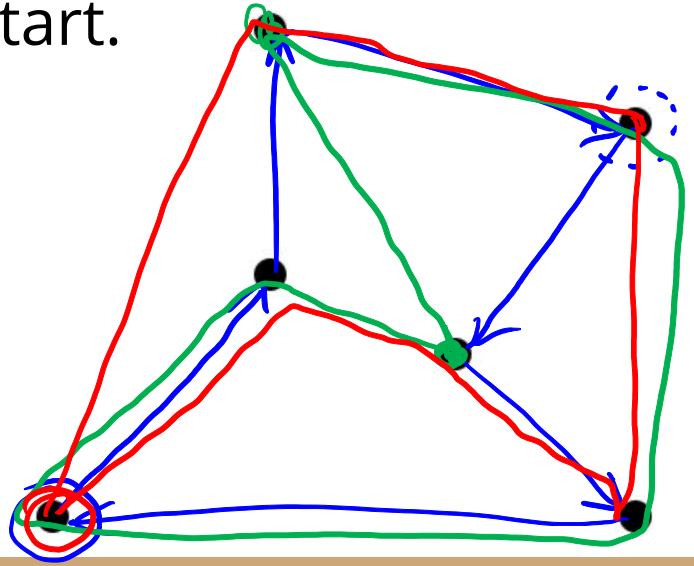
How many Starbucks are in Vancouver? 102Starbucks

Running Errands

SSSP Single Source Shortest Path

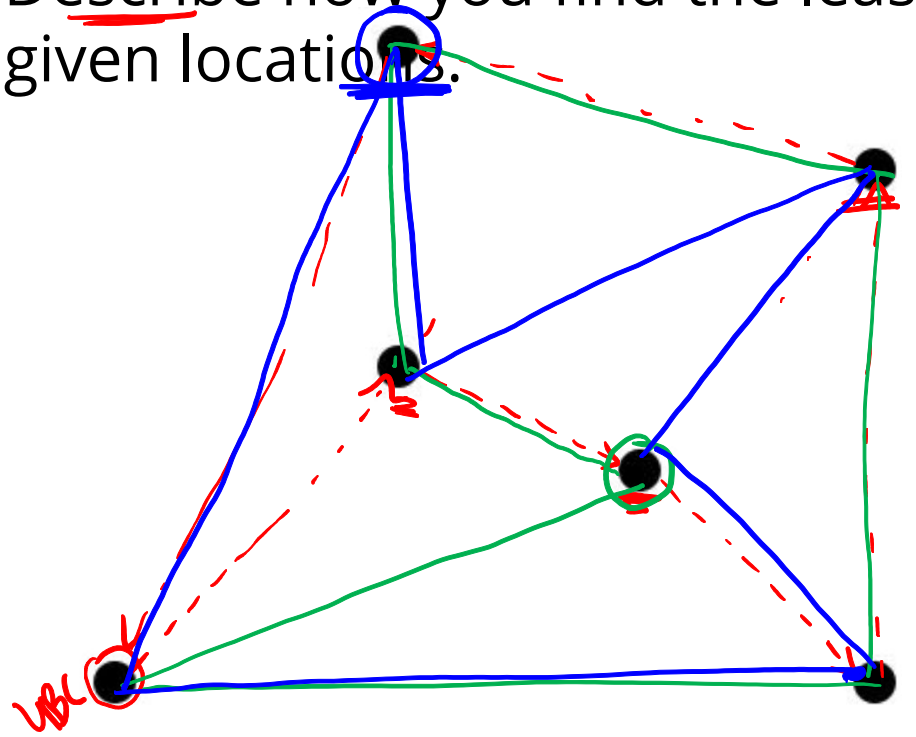


Determine the least cost route through a set of given locations, returning to the start.



Running Errands

Describe how you find the least cost route through a set of given locations.



Start someplace (strategically?)

① Repeat: go to closest unvisited place.
a) sl t centrally

② Repeat: go to farthest from start. remaining

Traveling Salesperson Problem (TSP)

One of the most well-studied problems in computational mathematics.

No algorithm works on all input configurations.

What does "works" mean? Finds shortest route in a reasonable amt of time.

Traveling Salesperson Problem (TSP)

Most common approach to computationally infeasible problems:

Sacrifice optimality for feasibility --

- Heuristic – algorithms that return a route just maybe not best one. Every step makes a reasonable choice.
- Approximation – takes a heuristic & proves it's close to optimal solution.

TSP how many routes?

Suppose you have 5 locations. How many different candidate solutions are there? Generalize to k locations?

$10 = \frac{9!}{2} = 181440$

ABCDE
ABCED
ABDCE
ABDEC
ABECD
ABEDC

ACBDE
ACBED
ACDBE
~~ACDEB~~
ACEBD
~~ACEDB~~
ADBCE
~~ADBEC~~
ADCBE

ABCDE

A → B, C, D, E

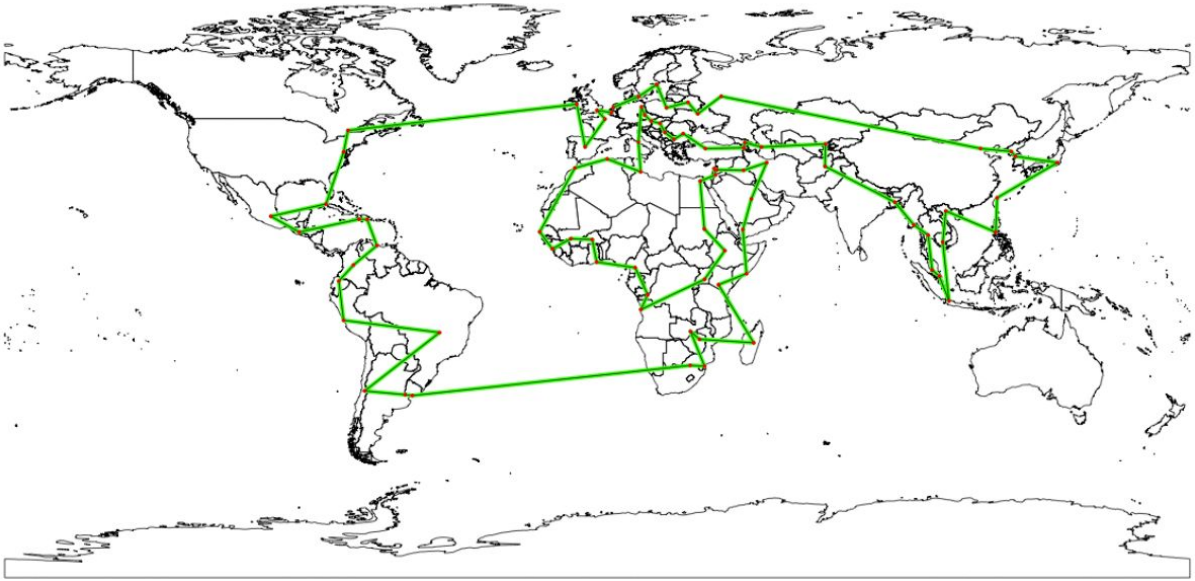
$4 \cdot 3 \cdot 2 \cdot 1$ Diff routes

$\frac{24}{2} = 12$

AEDCB

Demo Blog

https://towardsdatascience.com/around-the-world-in-90-414-kilometers-ce84c03b8552



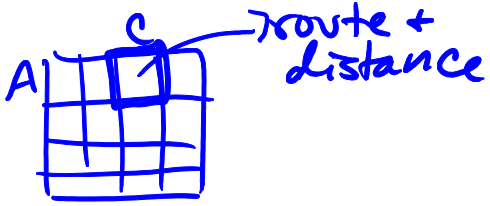
Plan for Code

Data available:

Given loc names → osmnx → lat/long

Given lat/long of 2 pts → networkX gives shortest dist + path.

Steps to assemble a solution:



1. Assemble distances betw loc into a table
2. List of all possible routes through locations (free!)
3. Compute min route among all possible
4. Put it all on a map.

Map applications

Three parts:

1. Assembling the data - OSM, local data stores, statsCan, etc. This is mostly the art of assembling geodataframes.

2. Computing on the data - osmnx simplifies graph algorithms and computation, but also supports other spatial computation.

3. Visualizing the data - matplotlib for static maps, folium for interactive maps.

POTD #38 Tue

https://github.students.cs.ubc.ca/cpsc203-2019w-t1/potd36

Describe any snags you run into:

1. Line ___: _____
2. Line ___: _____
3. Line ___: _____
4. Line ___: _____
5. Line ___: _____

ToDo for next class...

POTD: Continue every weekday! Submit to repo.

Reading: TLACS Ch 10 & 12 (lists and dictionaries)

References:

https://www.youtube.com/watch?v=wsSEKm-rU6U

https://github.com/gboeing/osmnx-examples/tree/master/notebooks

https://gist.github.com/psychemedia/b49c49da365666ba9199d2e27d002d07