Project 1: Travelling Salesperson Problem - Brute Force

- Learning objectives. At the completion of this project, you should be able to:
  - Implement a method to generate all permutations of a set of numbers.
  - Trace a Hamiltonian path through an undirected graph.
  - Use brute force to find the minimum cost solution to a Traveling Salesperson Problem.

- Background
  - A Traveling Salesperson Problem (TSP) is an NP-complete problem. A salesman is given a list of cities and a cost to travel between each pair of cities (or a list of city locations). The salesman must select a starting city and visit each city exactly one time and return to the starting city. His problem is to find the route (also known as a Hamiltonian Cycle) that will have the lowest cost.

- Problem
  - You will be expected to use brute force to calculate the minimum cost paths for a series of problems.
  - Data for each problem will be supplied in a .tsp file (a plain text file).

- Hints
  - Be careful of exponential explosion. 10 cities will have over 3 million possible paths.
  - The largest data set will be 12 cities.
  - I strongly urge you to create reusable code. It will be needed for the future projects.
  - In the future projects, we will be dealing with MUCH larger datasets. So large that brute force approach would not be possible.

- Deliverables
  - Project report (2-3 pages). Describe how you generated the permutations representing the tours. Show the route and the cost of the optimal tour for each provided dataset. Report run-times for each problem set.
  - Well-commented source code for your project (You can use any language you like, but I reserve the right to ask you to demo performance of your algorithm on a new dataset).
  - You don't have to include a GUI with visual representation of the solutions for this project, but it might be useful for your future TSP related projects in this course.

Data Format: You can read about standard TSP problem files here:
http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/

Data files for development and testing of your code will be generated using Concorde:
https://www.math.uwaterloo.ca/tsp/concorde/index.html

Sample data file with 7 cities:
NAME: concorde7
TYPE: TSP
COMMENT: Generated by CCutil_writetsplib
COMMENT: Write called for by Concorde GUI
DIMENSION: 7
EDGE_WEIGHT_TYPE: EUC_2D
NODE_COORD_SECTION
1 87.951292 2.658162
2 33.466597 66.682943
3 91.778314 53.807184
4 20.526749 47.633290
5 9.006012 81.185339
6 20.032350 2.761925
7 77.181310 31.922361

Distance Formula: $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$