City Block Coding Challenge

You are given a pseudo-city map. The map is full of odd rules one-way streets, dead-ends and other pitfalls. Your goal is to navigate these streets and get to your destination.

What is given

- Base 2D city map (x,y)
  - given by file (**grid_moves.pickle**) with a dictionary of allowed moves/transitions at each grid point
  - each point in the grid is a key with the values being the allowed transitions out of that point e.g. *"(2,2) : [(0,-1)]"* would denote that from point *(2,2)* the only available transition is to point *(2,1)*
  - note that allowed transitions are relative to grid point
- Your starting point on the grid and your destination point
  - `start_point = (5,3)`
  - `end_point = (0,7)`

Requirements

- Create a python script that does the following
  1. reads in grid_moves.pickle
  2. Finds all unique valid paths between given start_point and end_point
     - A valid path is one that follows the allowed moves from grid_moves dictionary and one that does not visit the same grid point twice
  3. Right out all above paths to **all_paths.txt** where each row is a unique path
     - The format for each row/path is

       `[(start_point_x,start_point_y),(step1_x,step1_y),(step2_x,step2_y),...,(end_point_x,end_point_y)]`

Submission guidelines

- Email a zip file with the following components
  - Your python code
  - grid_moves.pickle (input file)
  - all_paths.txt (output file)