Your task is to build an agent to solve a modifed version of the 8 puzzle problem (called the Expense 8 puzzle problem). The task is still to take a 3X3 grid on which 8 tiles have been placed, where you can only move one tile at a time to an adjacent location (as long as it is blank) and figure out the order in which to move the tiles to get it to a desired configuration. However now the number on the tile now also represents the cot of moving that tile (moving the tile marked 6 costs 6).  
  
Your program should be called expense\_8\_puzzle and the command line invocation should follow the following format:  
  
*expense\_8\_puzzle.py <start-file> <goal-file> <method> <dump-flag>*

* <start-file> and <goal-file> are required.
* <method> can be
  + bfs - Breadth First Search
  + ucs - Uniform Cost Search
  + dfs - Depth First Search
  + dls - Depth Limited Search (Note: Depth Limit will be obtained as a Console Input) [Note: This part is EC for CSE 4308 students]
  + ids - Iterative Deepening Search [Note: This part is EC for CSE 4308 students]
  + greedy - Greedy Seach
  + a\* - A\* Search (Note: if no <method> is given, this should be the default option)
* If <dump-flag>  is given as true, search trace is dumped for analysis in trace-<date>-<time>.txt (Note: if <dump-flag> is not given, assume it is false)
  + search trace contains: fringe and closed set contents per loop of search(and per iteration for IDS), counts of nodes expanded and nodes

Both start file and goal file need to follow the format as shown here:

Start file

2 3 6

1 0 7

4 8 5

END OF FILE

Goal file

1 2 3

4 5 6

7 8 0

END OF FILE

Your output needs to follow the format given in the example here:  
  
For:  
  
*expense\_8\_puzzle.py*[*start.txt*](https://crystal.uta.edu/~gopikrishnav/classes/2023/spring/4308_5360/assmts/assmt1_files/start.txt)[*goal.txt*](https://crystal.uta.edu/~gopikrishnav/classes/2023/spring/4308_5360/assmts/assmt1_files/goal.txt)*a\* true*  
  
The output should appear as follows:  
  
*Nodes Popped: 97  
Nodes Expanded: 64  
Nodes Generated: 173  
Max Fringe Size: 77  
Solution Found at depth 12 with cost of 63.  
Steps:  
        Move 7 Left  
        Move 5 Up  
        Move 8 Right  
        Move 7 Down  
        Move 5 Left  
        Move 6 Down  
        Move 3 Right  
        Move 2 Right  
        Move 1 Up  
        Move 4 Up  
        Move 7 Left  
        Move 8 Left*

Note: for both greedy and A\* search you need to come up with a acceptable heuristic .