Queens College of CUNY
Department of Computer Science
Design and Analysis of Algorithms (CSCI 323)
June 2024

Assignment #7 "Graphs and Graph Algorithms"

Due: June 20, 2024

Overview:

In this assignment, we implement some general infrastructure for reading, generating, representing, tabulating, and drawing graphs. We will use this in subsequent assignments that compare different algorithms for a shared problem.

Submissions:

In the Google form, please submit the following:

- Assignment7.py (source code)
- Assignment7.txt (console output)
- Assignment7-times.png (bar graph of timings)
- Assignment7-graph.png (graph of points and path)

Tasks:

Follow the template and general guidelines for previous assignments. Wherever possible, import those assignments and invoke their functions rather than creating and maintaining copies of the same code.

[1] Define a function read_graph(file_name) that reads a graph from a text file and returns an adjacency/cost matrix.

[2] Define a function adjacency_table(matrix) that accepts a graph as an adjacency/cost matrix and returns an adjacency/cost table.

[3] Define a function edge_set(matrix) that accepts a graph as an adjacency/cost matrix and returns an edge/cost set.

[4] Define a function edge_map(matrix) that accepts a graph as an adjacency/cost matrix and returns an edge/cost set.

[5] Define a function zero_to_inf(matrix) that replaces non-edges represented by a cost of zero (0) with a cost of INF, tentatively set at 9999.

[6] Define a function random_graph(size, max_cost, p=1) that generates a graph with *size* edges, where each edge (except loops/self-edges) is assigned a random integer cost between 1 and *max_cost*. The additional parameter p represents the probability that there should be an edge between a given pair of vertices.

[7] Define functions that traverse the graph in these two standard orderings
- Breadth-First Search (BFS)
- Depth-First Search (DFS)

[8] Define a function print_graph(matrix) that prints a graph in matrix form.

[9] Define a function print_graph_tabular(matrix) that prints a graph in tabular form.

[10] Define a function draw_graph(graph) that draws a graph

See https://stackoverflow.com/questions/20133479/how-to-draw-directed-graphs-using-networkx-in-python

[11] Define functions for various graph-related tasks using the different models and compare them by tabulating and plotting their runtimes..

```
# Define a function draw_graph(graph) that draws a graph and saves it as a file.
#
https://stackoverflow.com/questions/20133479/how-to-draw-directed-graphs-using-networ
kx-in-python
# https://stackoverflow.com/questions/74312314/draw-a-directed-graph-in-python
def draw_graph(edges, directed, filename):
    G = nx.DiGraph()
    G.add_edges_from(edges)
    val_map = {'A': 1.0, 'D': 0.5714285714285714, 'H': 0.0}
    values = [val_map.get(node, 0.25) for node in G.nodes()]
    pos = nx.spring_layout(G)
    cmap = plt.get_cmap('jet')
    nx.draw_networkx_nodes(G, pos, cmap=cmap, node_color=values, node_size=500)
    nx.draw_networkx_labels(G, pos, font_size=12, font_color='white')
    if directed:
                   nx.draw_networkx_edges(G,  pos,  edgelist=edges,  edge_color='r',
arrows=directed, arrowsize=10)
    else:
        nx.draw_networkx_edges(G, pos, edgelist=edges, edge_color='r', arrows=False)

    plt.savefig(filename)
    plt.show()
```