Programming Assignment IX
Shortest Path

Due Date: 2024/01/16 (60)

1 Description of Assignment

Design and implement efficient data structure and algorithm for computing shortest path from the source vertex s to destination vertex t in a weighted graph. Dijkstra's algorithm and priority queue data structure is preferred in this assignment.

2 Input

The input file is a weighted graph $G = (V, E, w)$ and a source vertex s. For the input of the graph, refer to the sparse matrix in Programming Assignment III.

3 Output

The output is a shortest path tree from s. Print the shortest path tree by listing its edges. Graph visualization libraries can also be used to display the graph and the shortest path. If there is no graph visualization library available in your programming environment, it is also acceptable to draw it manually in your report.

Notes

The format of the report of the assignment should be close to the format of a research technical report. Incluse at least the following in your report.

1. Title and Author.
   This section should include assignment number, *your name*, *student number* and *email address* on the *first* page of your report.

2. Statement of the problem.
   A "formal" description of the problem in this assignment. In addition to the basic requirements specified in the assignment, highlight other functions or features that you have implemented.

3. Main results.
   This section should include at least the following items.

   (a) Description of the design of your program.

(b) Describe the data structures used in the program to improve the efficiency of the program. These data structures should be implemented by you and appear in the first part of your program.

(c) List of your program with comments.

i. If your program is very long, list only the main parts of the program here and the entire program in appendix.

ii. Additional comments can be added manually to explain the design of the program.

(d) Outputs of the compilation and the executions of your program.

4. Conclusions
   Give a brief summary of what you did, and interesting thing you learned from this assignment.

Additional notes:

1. Turn in your report on or before due day.

2. The output of the program execution should indicate the correctness of your program. In other words, a set of comprehensive (but not necessarily exhaustive) annotated test data for the problem should be provided to show that your program is indeed correct. This can be done by carefully selecting a set of test data.

3. Print or write the report on A4 papers. Bind them together in the upper left corner.