Australian National University
Department of Computer Science

Computer Science COMP3600 in 2002 – Assignment 3

Due: 5pm, Wednesday, Oct. 23 **Late Penalty:** 25% per day

Question 1.(35/35)

Given integers n and m, construct an undirected weighted random graph generated by the following pseudo-code, where n is the number of vertices and m is an upper bound of the edges in the graph.

```
Generate_graph(V, E)
    V ← {1, 2, ..., n}; E ← ∅
    count ← 0 /* the number of the edges in the graph */
    get the current time as the random seed;
    repeat
        choose two vertices in the graph at random, e.g., u and v
        if there is no edge between u and v then
            add an edge between u and v ,
            i.e., E ← E ∪ {(u, v)};
            assign the edge with a random integer between 1 and 100
        endif;
        count ← count + 1
    until (the graph is connected) and (count == m).
```

If $count = m$ but the graph is disconnected, then you should use a different random seed to run the above procedure again until the resulting graph is connected and $count = m$. Having the graph G, the remaining tasks are as follows.

(a) print the generated graph when $n = 12$ and $m = \lceil 1.2n \log n \rceil$ by listing the edges in G, the generation of the graph worths 5 points. (7/35)

(b) Use Kruskal's algorithm to find a minimum spanning tree (MST) in G. (8 points, 8/35)

(c) Use Prim's algorithm to find a minimum spanning tree (MST) in G. (8 points, 8/35)

(d) Assume that $n = 13$ and 120, for each different n, consider the three upper bounds of the number of edges in the graph with $m = \lceil 1.5 * n \log n \rceil$, $m = \lceil n^{4/3} \rceil$, and $m = \lceil n^2/8 \rceil$. List the real running times of the above two algorithms for each combination of n and m, where the timing of an algorithm is starting from running the algorithm until it finished. It neither includes the time for construction of the graph nor including the time for printing the MST. Through examining the running time and the problem size for these two algorithms, what kind of conclusions you can get? (12 points, 12/35)

What to submit.
The program you submitted should take two command-line parameters: the number of vertices and the number of edges. In other words, the command should be

`mst` n m

You can submit the following files and document. Note that you should use the exact names in the following list.

- `makefile`
- `mst.c`
- `a3_report.ps` (optional)

where the file `makefile` is compulsory. The shell command **`make mst`** must compile and links to the program `mst`. Your program must contain full documentation. You may not need to write a separate report, but your program must contain full documentation. Otherwise, you may lose marks.

Question 2 (bonus question) (extra 5 points)
Warning: If you cannot finish Question 1, you are NOT expected to finish the following problem.

Given an undirected weighted graph $G(V, E)$ with n nodes and m edges and an integer Δ, associated with each edge $e = (u, v) \in E$, there are two weights $c(e)$ and $\delta(e)$, which are the cost and the delay of e, the *delay-constrained shortest path problem* in G between node s and node t is to find a simple path P between s and t such that (i) $\sum_{e \in P} c(e)$ is minimized; and (ii) $\sum_{e \in P} \delta(e) \leq \Delta$. If there are several shortest paths with the same minimum cost, choose the one with the minimum delay only.

Devise an algorithm for this problem and analyze the time complexity of the proposed algorithm. No coding for this problem is required. But you need to give the basic steps of your proposed algorithm and the detailed analysis.