CS483 - Practice Problems (due Oct 13)

Jana Košecká

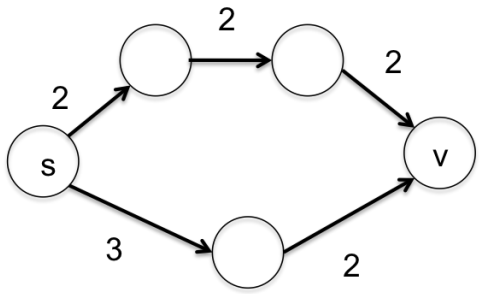
Chapter 3

1. (6) Problem 3
   We will run the same topological ordering algorithm as in the textbook. If at every iteration we can find a node with no incoming edges we will get the desired topological order. If at least in one iteration every node has at least one incoming edge, when we say in the proof in the textbook that there is a cycle. So this would be essentially a way how to find a cycle in $O(m+n)$ time.

Chapter 4

1. (6) Problem 2
   a) True. Since the sorted orders of the edges costs will be the same, Kruskal's algorithm will return the same Minimum weight spanning tree.

   b) False. Consider squaring the costs of this graph and the shortest path will change.



Chapter 5

1. (7) Problem 1 This problem can be solved using divide and divide and conquer. A and B are the two databases and i^th is the smallest element of A and B. First lets compare the medians of the two databases. Let k be $\frac{1}{2}n$ then $A(k)$ and $B(k)$ are two medians. If $A(k) < B(k)$ then $B(k)$ is always greater then first k-1 elements of B, it is at least $2k^{th}$ element in the combined database. Since $2k \geq n$, all elements are greater than $B(k)$ are greater then median, so we can eliminate the second half of B. This will help you decide which part of the array to look next and which to eliminate, you will then apply the same recursive strategy to the part which is left. The recurrence for the running time is then

$$T(n) = T(n/2) + 2 = O(\log n)$$

   The running time - the solution can be found using the recurrence tree.