Ann, Bob, Charlie (*replace with your names*)
COSC 336
3/19/2020 (*replace with the current date*)

Assignment 8 - Solutions

Exercise 1.

We have seen that Dijkstra's algorithm can be implemented in two ways: Variant (a) uses an array to store the $dist[]$ values of the unknown nodes, and Variant (b) uses a MIN-HEAP to store these values.

(a) Suppose in your application $m \leq 3n$. Which variant gives a faster runtime? Justify your answer.

(b) Suppose in your application $m \geq n^2/3$. Which variant gives a faster runtime? Justify your answer.

(c) Suppose that your application $m = n^{3/2}$. Which variant gives a faster runtime? Justify your answer.

Solution: The runtime of the array implementation (variant (a)) is $\Theta(n^2 + m)$, and the runtime of the Min Heap implementation (variant (b)) is $\Theta(m \log n)$.

Therefore, if we plug the given value of m in the above formulas, we conclude:

(a) If $m \leq 3n$, the runtime for the array implementation is $\Theta(n^2 + 3n) = \Theta(n^2)$, and the runtime of the Min Heap implementation is $\Theta(3n \log n) = \Theta(n \log n)$. So, the Min Heap implementation is faster.

(b) If $m \geq n^2/3$, the runtime for the array implementation is $\Theta(n^2 + n^2/3) = \Theta(n^2)$, and the runtime of the Min Heap implementation is $\Theta(n^2/3 \log n) = \Theta(n^2 \log n)$. So, the array implementation is faster.

(A more formal argument is that for the array implementation the runtime is at most $O(n^2 + n^2) = O(n^2)$, because m cannot be larger than n^2, and for the Min Heap implementation, the runtime is at least $\Omega(n^2 \log n)$.)

(c) If $m = n3/2$, the runtime for the array implementation is $\Theta(n^2 + n^{3/2}) = \Theta(n^2)$, and the runtime of the Min Heap implementation is $\Theta(n^{3/2} \log n) = \Theta(n^{3/2} \log n)$. So, the Min Heap implementation is faster.

□

Exercise 2. Recall that when we do DFS with timing every node u gets 2 numbers that were denoted $u.d$ and $u.f$. $u.d$ is the discovery time and $u.f$ is the finish time.

Show that in a DAG (directed acyclic graph), for any two nodes u and v such that there exists a path from u to v, it holds that $u.f > v.f$.

Hint: There are two cases to analyze. Case 1 is that $u.d < v.d$ (in words, u is discovered before v), and Case 2 is that $v.d < u.d$ (so, v is discovered before u). In both cases, you need to argue that $u.f > v.f$.

Solution: If u is discovered before v, since there is a path from u to v, node v is discovered during the exploration of u (in other words, while u is *gray*, using the terminology from the textbook), and so v is finished before u is finished.

If v is discovered before u, since there is no path from v to u (because, otherwise there would be a cycle, which is not possible in a DAG), v is finished before u is even discovered, and so clearly u is finished after v is finished.

□