KØBENHAVNS UNIVERSITET

Assignment 5

Alex (zhc522), Adit (hjg708), Usama (mhw630), Nikolaj (bxz911)

AD 2023

10. april 2023

Task 1

Prove that an entry H_{ij} of minimal value must be an edge of T – i.e. if H_{ij} has minimum value among all entries of H, then (i, j) is an edge of T and has $w(i, j) = H_{ij}$; the minimum is only taken over non-diagonal entries of H. Do not mention minimum spanning trees in your proof.

Hint: Try to prove this by contradiction. Recall that all edges of G have positive weight (strictly greater than 0).

We have to prove that an entry H_{ij} of minimal value must be an edge of T. To prove this we will use contradiction. Let us hypothesis that an entry of minimal value is not an edge, but a path made out of edges. For that to happen we should add two or more paths together, to get the minimal value of an entry H_{ij}. That would mean that one or more edges would have to have a negative value, for that to be the case. Since we know that each edge has a positive value strictly greater than 0, then this property would not hold. Therefore an entry H_{ij} of minimal value must be an edge of T.

Task 2

Consider the complete graph $G(H)$ as described in the example above. Prove that a minimum spanning tree T' of $G(H)$ is equal to T. You must prove the statement for the general case and not just the example.

Hint: Consider an edge e added in a step of Prim's or Kruskal's algorithm. Is it possible that e was not a part of T? You should be able to argue the same way as in task 1.

As we assume that T' is not a MST, meaning that $T \neq T'$. And we let $e = (u, v)$ be the first edge chosen by Prim's algorithm which is not in T. We let T' be the tree we have before we add in e, and let T be the MST.

Now, we suppose that we add e to T', then Prim's algorithm would choose a random node and pick the edge with the smallest value between another node. If e is inserted in T', then depending on the weight of e could reshape T'. Cycles are not allowed in MST. We know $w(e') \geq w(e)$ by definition of the algorithm. So if we added e to T' and remove e', we get a new tree while preserving minimal total weight of T'. This new tree has the same number of edges as T', and the total weights of its edge e is not larger that that of T. This process when repeated indefinitely will show that T' is equal to T, and this shows that the tree that is generated by any instances of Prim's algorithm is a minimal spanning tree.

Task 3

What is the running time of the algorithm resulting from running a MST algorithm on the input and returning the list of edges as a function of n? (Note that $|E(G(H))| = \theta(n^2)$).

The number possible routes (combination) in $G(H)$ is $n(n-1)$. But we can shorten this calculation by not recounting the same combinations again and again. So $\frac{n(n-1)}{2} = \frac{n^2-n}{2} = \Theta(n^2)$ are the possible edges. In CLRS page 636 it says the running time of Prim's algorithm is $O(E\,lgV)$, where E is the number of edges and V is number of vertices equal to n. Therefore the upper bound running time is $O(n^2\,lg\,n)$