1 Dynamic Shortest Path on Grid

Consider a directed graph that’s a $r \times c$ grid. Show that we can support the following two operations in $O(c^3 \log r)$ time each:

1. Modify edge length.
2. Given two query vertices $s$ and $t$, return the length of the shortest path between them

2 Dynamic Shortest Path on General, Directed Graphs

Suppose there is a data structure that supports each of the following on a general, dynamic graph in time $\delta$:

1. Modify edge length.
2. Given two query vertices $s$ and $t$, return the shortest path between them with $O(l)$ overhead if the path has length $l$.

Show that this can be used to give an $O(n\delta + n \log n)$ time algorithm for bipartite matching.

3 Re-rooting Link Cut Trees

The link-cut trees that we described so far have a fixed root per subtree. Show that an additional operation that changes this root can be supported in $O(\log n)$ time.

4 Counting Interiors of Cycles

Show that we can preprocess a planar graph with a given embedding such that given a cycle in this graph of length $l$, we can return the number of vertices contained inside this cycle in $O(l)$ time. Note: This problem does not require the use of planar separators. Bonus: Give an algorithm with $\tilde{O}(n)$ preprocessing time.

5 Ancestor Query

Build a data structure that processes a tree in $O(n)$ time and answers the following query in $O(1)$ time: given two vertices $u$ and $v$, is $u$ an ancestor of $v$. 
6 Computing Stretch
Recall that given a spanning tree $T$, the stretch of an off-tree edge equals to the length of the tree path connecting its endpoints divided by the length of the edge. Give a $O(m \log \log n)$ or faster algorithm for computing the stretch of all off-tree edges. hint: process things offline, and use union-find.

7 Min-cost Flow on a Line
We can augment the maximum flow problem with costs by charging each unit of flow through edge $e$ a cost of $c_e$. The minimum cost flow problem asks to find a flow from $s$ to $t$ routing $F$ units of minimum cost. Consider a directed graph on a line $v_1 \ldots v_n$ with capacities are costs. There are edges from $s$ to each $v_i$, each $v_i$ to $t$ and each $v_i$ to $v_{i+1}$. Show that a minimum cost $st$ flow which also routes the maximum amount of flow from $s$ to $t$ can be found in $O(n \log n)$ time.

8 Incremental 2-edge connectivity
Show that we can support inserting an edge and checking whether there are 2 edge-disjoint paths from $u$ to $v$ in $O(\log n)$ amortized time.

9 Offline Dynamic Minimum Spanning Tree
Given a sequence of $n$ inserting/deletion operations, show an algorithm that computes implicit representations of the minimum spanning trees in all iterations in $O(n^{3/2})$ total time.
bonus: Do the same in $O(n \log n)$ time.

10 Replacement Shortest Path on Undirected Graphs
Given a graph $G$ with $st$ shortest path $v_0 \ldots v_l$. The replacement shortest path for $i$ is the shortest $st$ path in $G \setminus v_i v_{i+1}$. Show that the length of all replacement shortest paths can be computed in $O(m \log n)$ time.