a) We need to show that there exists a min-cost arborescence which enters every 0-cost strongly connected component(ZSCC) exactly once. The proof is very similar to the proof done in Section 4.9 for cycles. Let T be a min-cost arborescence and for any ZSCC, S, let $e = (u, v)$ be the edge closest to the root, r, entering S. Now we delete all other edges entering S and edges (v_1, v_2) where $v_1, v_2 \in S$, and add edges by doing a DFS on S starting from v. Clearly the resulting graph is an arborescence since we have exactly one edge entering every vertex and every vertex is reachable from the root(for $w \in S$, w is reachable from v; for $w \notin S$ if the path to vertex w went through S with l being the last vertex on the path in S, we now have the path $r - v, v - l, l - w$). Also the cost of the new arborescence is no greater than the cost of T since we only added 0-cost edges. Therefore while contracting we can contract ZSCCs and while opening out we do a DFS to add edges.

b) We have $c''_e = \max(0, c_e - 2y_v)$ where $e = (u, v) \Rightarrow c_e \leq c''_e + 2y_v$. Therefore $c''_e = 0 \Rightarrow c_e \leq 2y_v$. Also $\sum_{v \neq r} y_v$ is a lower bound on $c(T_{opt})$ where T_{opt} is the min-cost arborescence with costs c_e. Since T has 0 c''-cost, we have, $c(T) = \sum_{e \in T} c_e = \sum_{e=(u,v), v \neq r} c_e \leq 2 \sum_{v \neq r} y_v \leq 2c(T_{opt})$.

c) We will prove this by induction on the no. of recursive calls we make. Let G^i, c^i, T^i, T^i_{opt} denote respectively the graph, cost function, arborescence constructed by the algorithm, and the min-cost arborescence(wrt. costs c^i) at the i^{th} stage(recursive call) of the algorithm. For an edge $e = (u, v) \in E^i$, we have, $y_v \leq c^{i-1}_e - c^i_e \leq 2y_v$. Suppose the algorithm terminates after k recursive calls. We will show by induction(on $k - i$ to be precise) that $c^i(T^i) \leq 2c^i(T^i_{opt})\ \forall i, 1 \leq i \leq k$. The base case is when $i = k$. So $c^k(T^k) = 0 \leq 2c^k(T^k_{opt})$. For the induction step assuming that $c^i(T^i) \leq 2c^i(T^i_{opt})$, we will show that $c^{i-1}(T^{i-1}) \leq 2c^{i-1}(T^{i-1}_{opt})$. Consider the arborescence T^{i-1}_{opt} with cost function c^i. We may modify T^{i-1}_{opt}(by deleting some edges and adding edges of 0 c^i-cost as in **a)**) so that it induces an arborescence, A of on greater c^i-cost on G^i. So we have, $c^i(T^{i-1}_{opt}) \geq c^i(A) \geq c^i(T^i_{opt})$ since T^i_{opt} is min-cost wrt. costs c^i. Now we have,

$$\begin{aligned}
c^{i-1}(T^{i-1}) &\leq c^i(T^{i-1}) + 2\sum_{v \neq r} y_v && (c^{i-1}_e \leq c^i_e + 2y_v)\\
&= c^i(T^i) + 2\sum_{v \neq r} y_v && (\text{since the edges added to } T^i \text{ all have 0 } c^i- \text{ cost})\\
&\leq 2(c^i(T^i_{opt}) + \sum_{v \neq r} y_v) && (\text{by the Induction Hypothesis})\\
&\leq 2(c^i(T^{i-1}_{opt}) + \sum_{v \neq r} y_v) && (\text{using the above lower bound})\\
&\leq 2c^{i-1}(T^{i-1}_{opt}) && (c^i_e + y_v \leq c^{i-1}_e)
\end{aligned}$$

and hence by induction $c(T) = c^0(T^0) \leq 2c^0(T^0_{opt}) = 2c(T_{opt})$ where T is the arborescence returned by the algorithm and T_{opt} is the optimal arborescence.

[1] ex271.554.851