Exercise 1: Tree Layouts

Let $T = (V, E)$ be a rooted binary tree. For a vertex $v \in V$, we denote its $x$-coordinate by $x(v)$ and its $y$-coordinate by $y(v)$.

(a) We draw the tree $T$ as follows. For each vertex $v$ of $T$, we set $x(v)$ equal to the rank of $v$ in a post-order traversal of $T$, and $y(v)$ equal to its depth in $T$.

(i) Show that the resulting straight-line drawing is planar.

(ii) What is the area of the drawing?

(iii) What happens if instead of a post-order traversal we use a pre-order traversal?

(iv) Can the algorithm be extended to rooted ordered trees?

(b) We draw the tree $T$ as follows. For each vertex $v$ of $T$, we set $x(v)$ equal to the rank of $v$ in a pre-order traversal of $T$, and $y(v)$ equal to the rank of $v$ in a post-order traversal of $T$.

(i) Show that the resulting straight-line drawing is planar and strictly downward (for each edge $(u,v)$, with $\text{depth}(u) < \text{depth}(v)$, it holds that $y(u) > y(v)$).

(ii) Show that a vertex $v$ is in the subtree rooted at vertex $u$ if and only if $x(v) > x(u)$ and $y(v) < y(u)$.

(iii) Do isomorphic subtrees have congruent drawings?

Exercise 2: HV-Layouts

Give an algorithm that for a given $n$-vertex binary tree constructs an HV-layout with minimum area in $O(n^2)$ time. Consider both ordered an non-ordered trees.
Exercise 3: Layouts of General Trees

Let \( T = (V, E) \) be an arbitrary rooted tree (i.e., not necessarily binary). Prove that a planar straight-line drawing \( \Gamma \) of \( T \) such that siblings (vertices with the same parent) have the same \( y \)-coordinate, parent-vertices are centered with respect to their children, and the area of \( \Gamma \) is in \( O(n^2) \), can be computed in \( O(n) \) time.

Exercise 4: Minimal-Width Layout

Let \( T = (V, E) \) be a rooted binary tree with a BFS-ordering and let \( \text{depth}(v) \) be the respective BFS-level of a vertex \( v \in V \). Formulate a linear program (LP) that computes a planar straight-line drawing \( \Gamma \) of \( T \) with minimal width such that \( \Gamma \) respects the BFS-ordering, parent nodes are centered with respect to its children, and each vertex \( v \) has \(-\text{depth}(v)\) as \( y \)-coordinate. Is the running time of the resulting algorithm polynomial in the size of \( T \)?