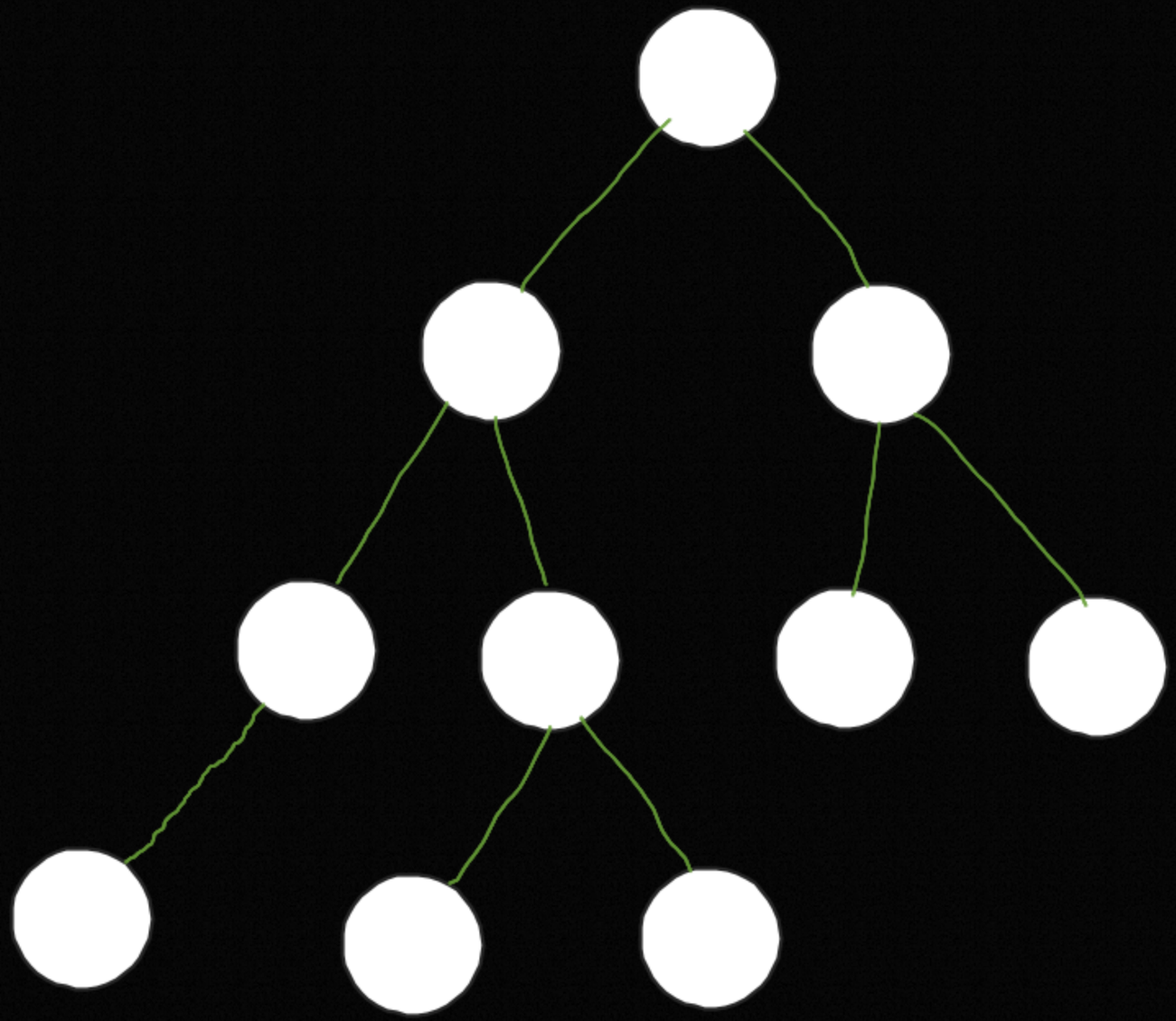
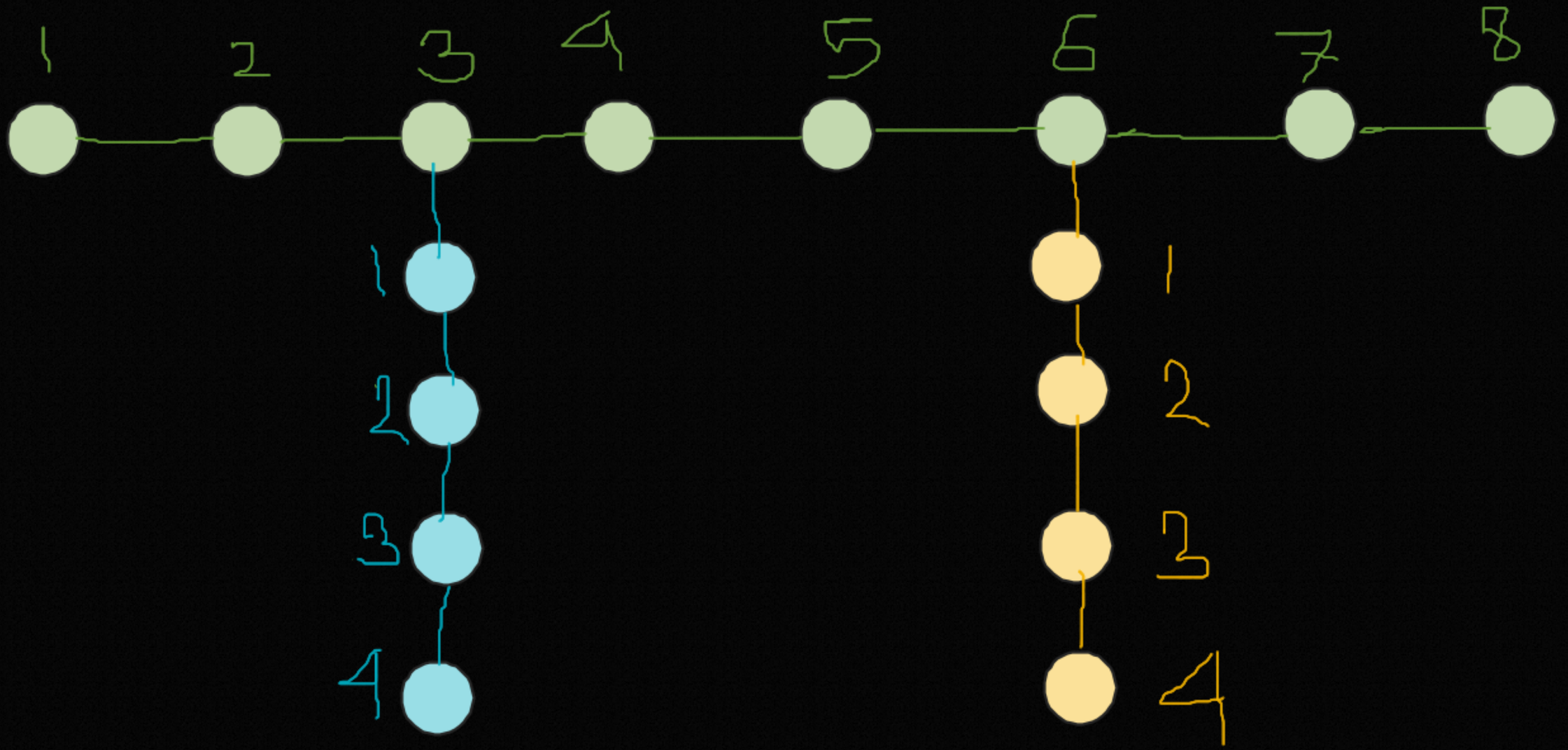
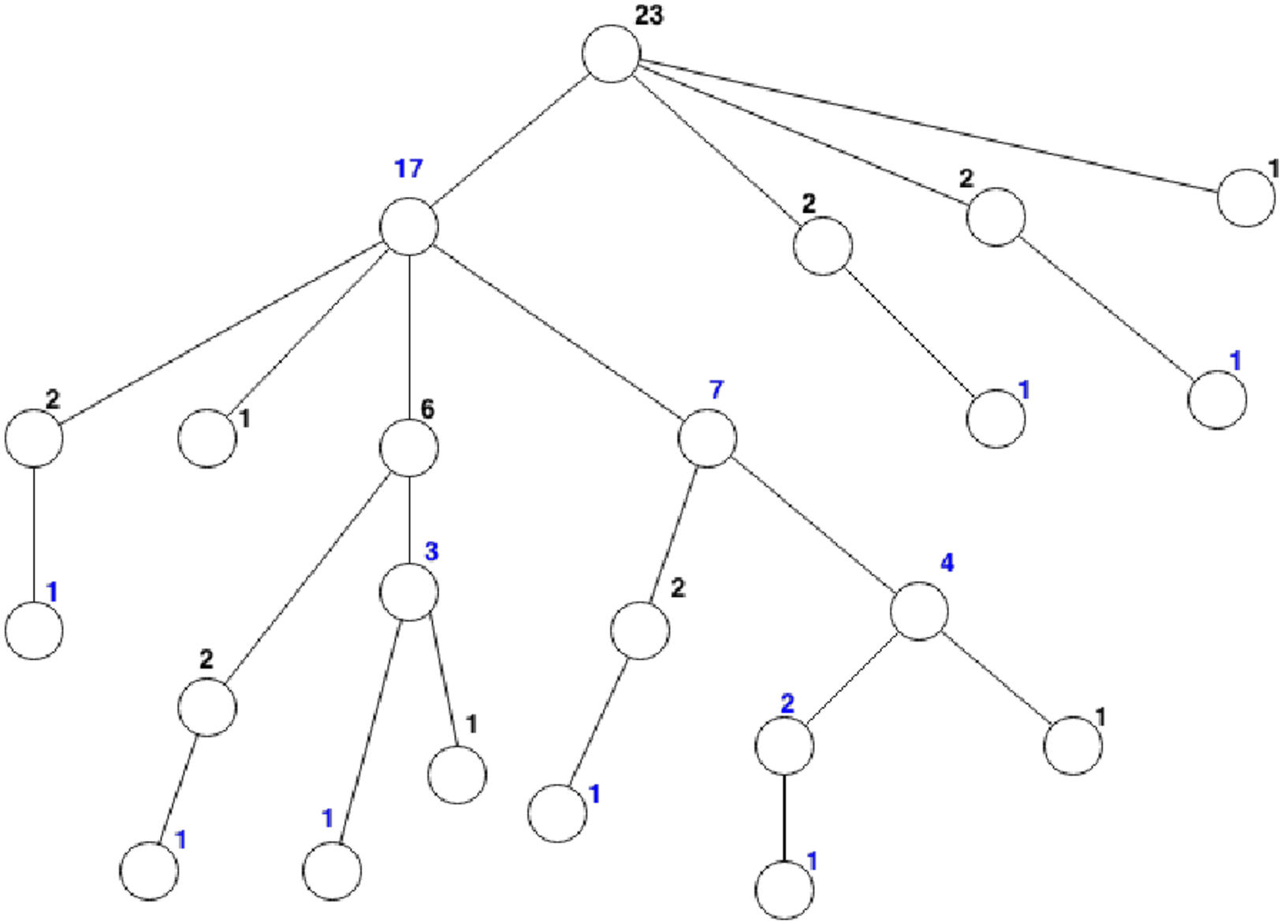
Heavy Light Decomposition

Types of Queries :

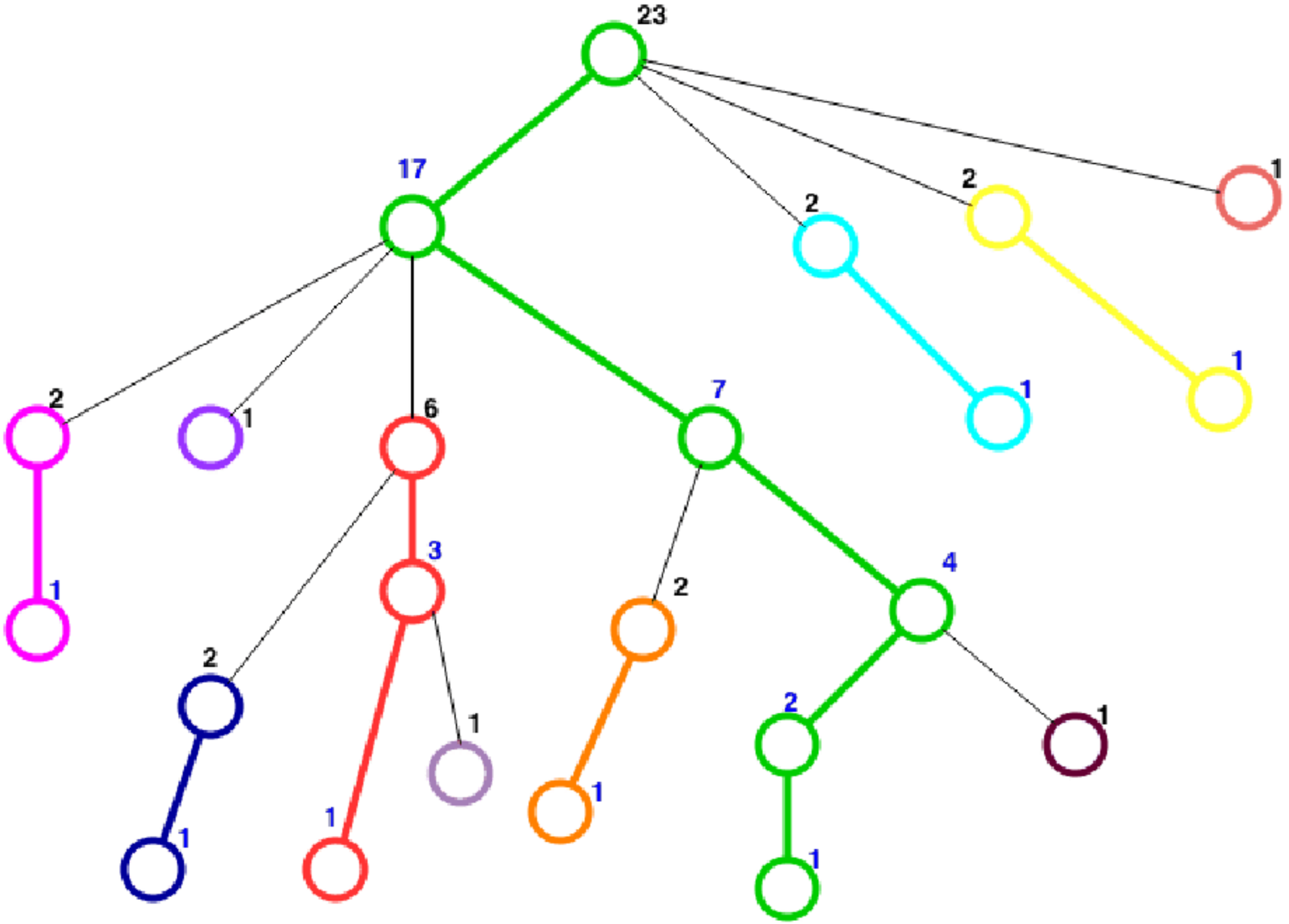
1. Sum/Max/Min value in Path(u,v)
2. Update all the nodes in Path(u,v)



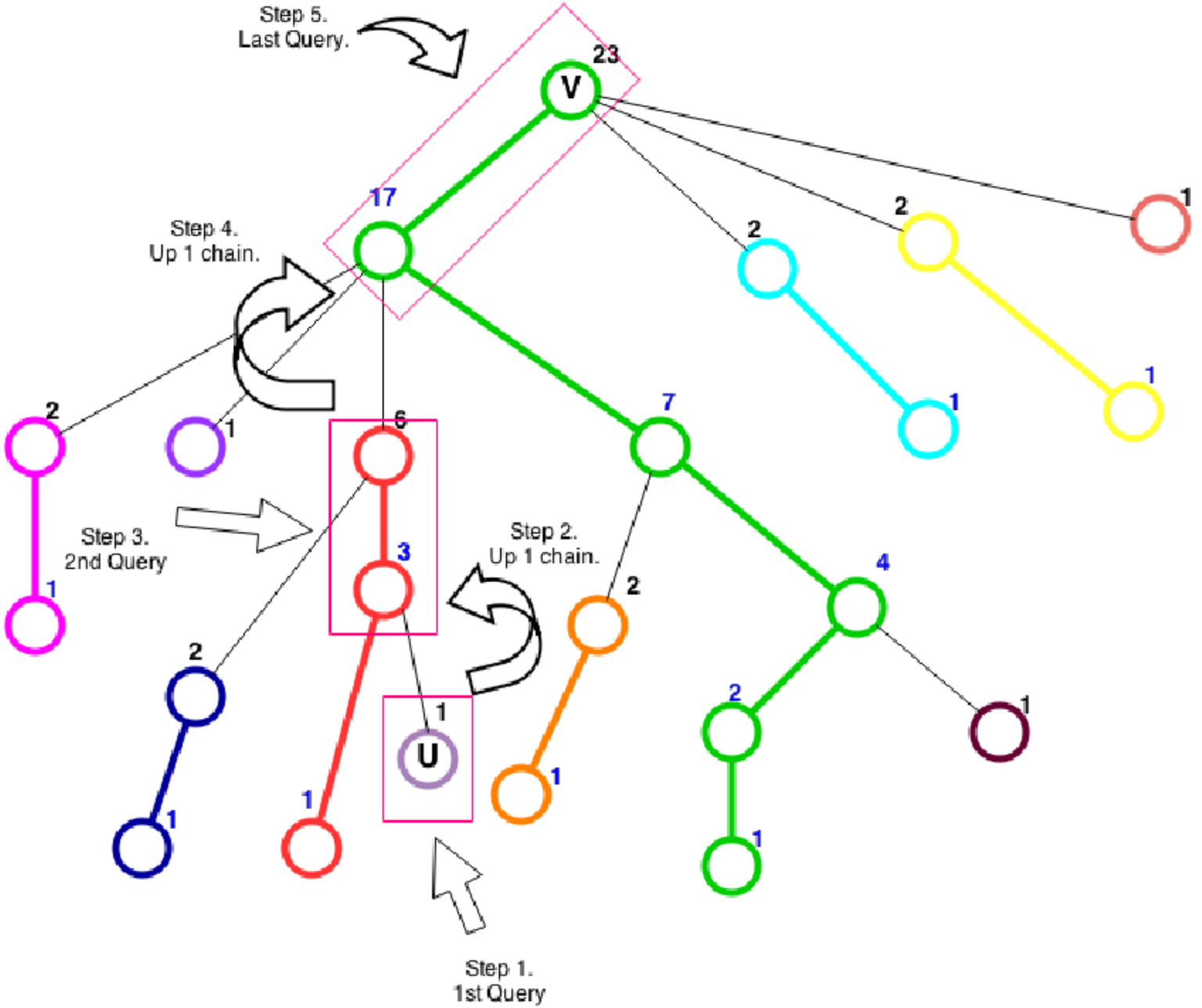

1
2
3
4
5
6
7
8
1
2
3
4
1
2
3
4



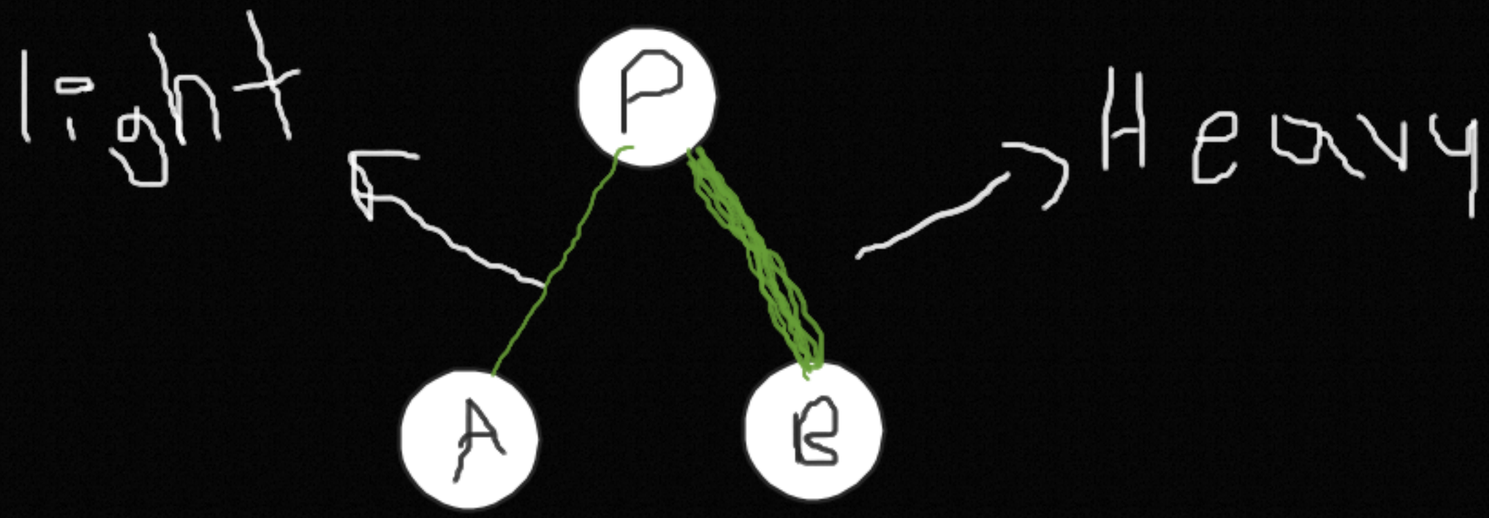
Each node has its sub-tree size written on top.
Each non-leaf node has exactly one special child whose sub-tree size is colored.
Colored child is the one with maximum sub-tree size.



Each Chain is represented with different color.
Thin Black lines represent the connecting edges. They connect 2 chains.

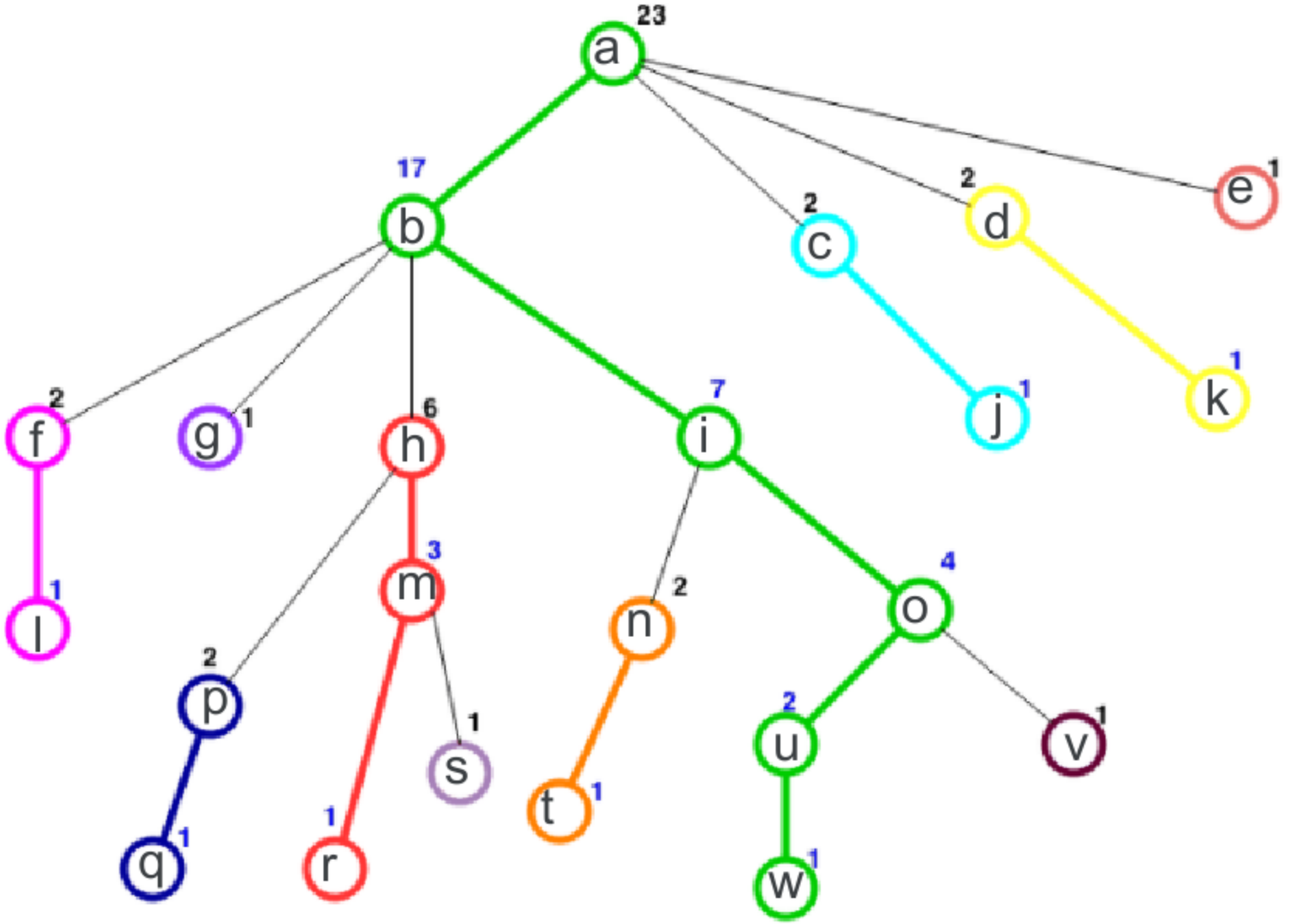


Consider the path from U to V.



size(B)>=size(A)
size(P)>=size(A)+size(B)

size(P)>=2*size(A)



a b i o u w---v---n t---c j---d k--e--f i---g---h m r---s--p q

Heavy Light Decomposition

- Break the tree into vertex-disjoint “chains” going from “top” to “bottom”
- For every node, the edge b/w max. size subtree child (**special child**) will be a “heavy” edge, rest will be “light” edges (**normal child**).
- Every light edge connects two different chains / a new chain starts after every light edge.

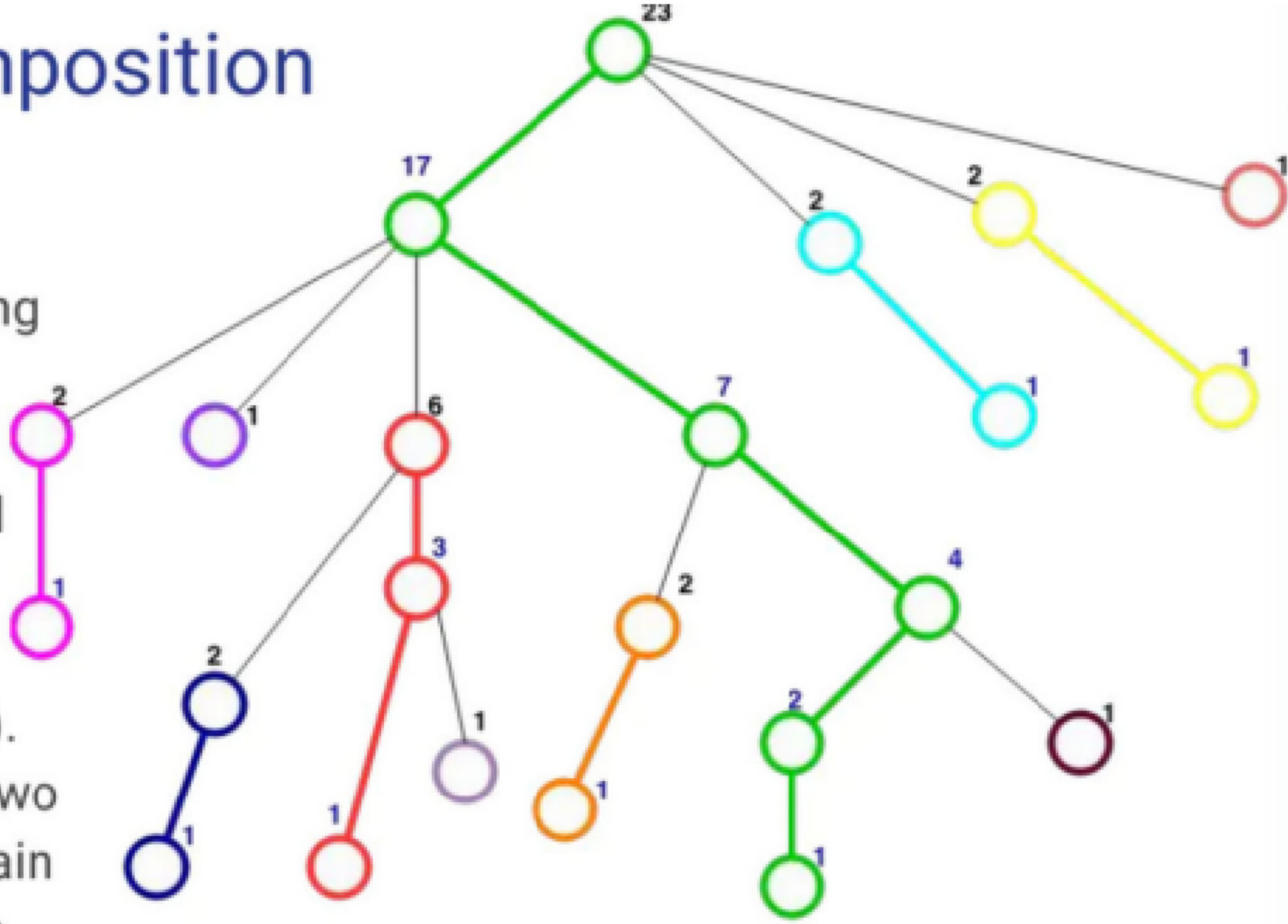


Image Source: https://blog.anudeep2011.com/heavy-light-decomposition/

HLD - Properties

- Every vertex is part of exactly 1 chain.
- Every chain forms a subarray in the "linearised" tree.
- Subtree size reduces by at-least half on traversing a "light" edge.
- Therefore, we can go up from any node **x** to it's ancestor node **p** by changing at-most logN chains.
- Any path A – B can be written as A – LCA + LCA – B; and hence can be traversed by changing at-most 2 * logN chains.

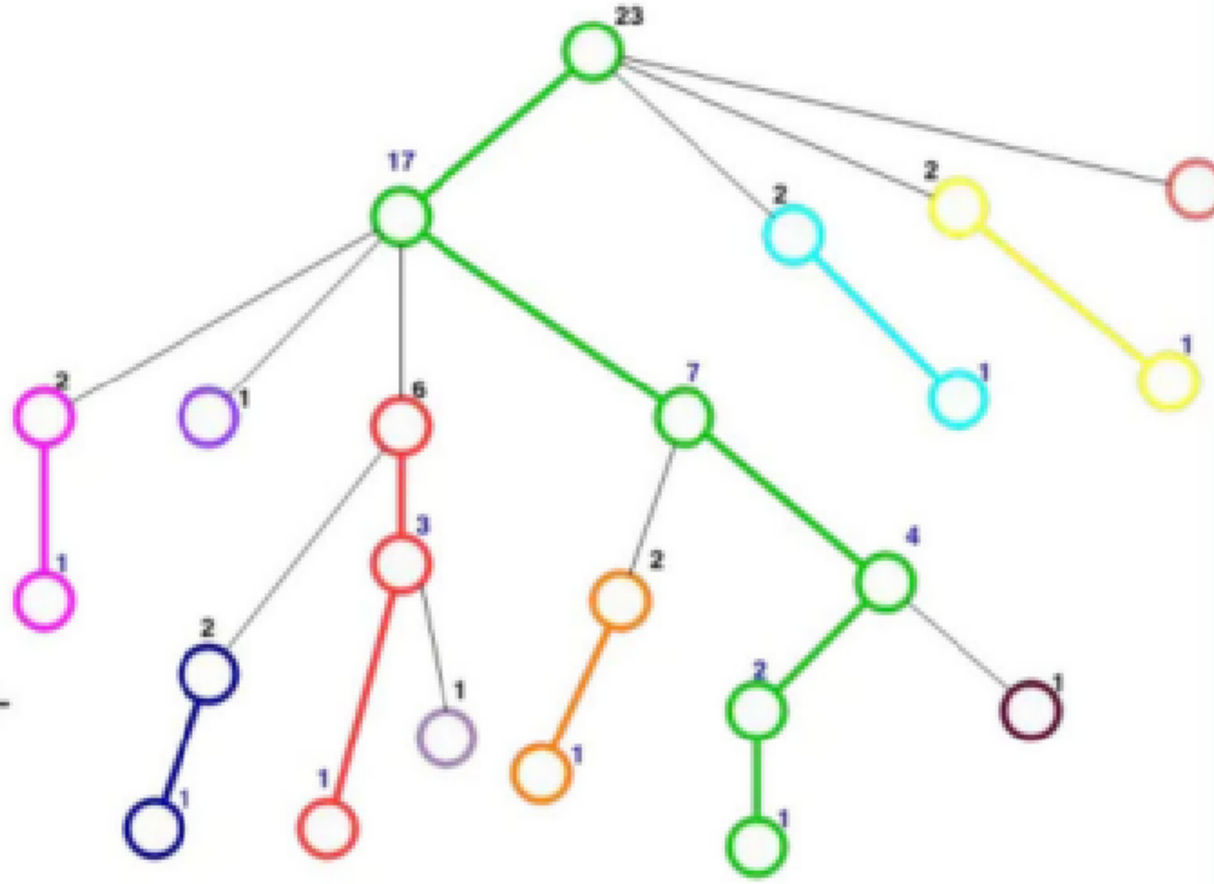


Image Source:
https://blog.anudeep2011.com/heavy-light-decomposition/

HLD - Steps to support path updates / queries

- Decompose the tree into chains via HLD.
- Linearise the chains into an array and build a Data Structure on the array that supports range queries / updates.
- For any path query/update b/w nodes A & B; process it as a query/update on O(logN) different ranges in the linearised array – corresponding to O(logN) chains that we need to traverse while going from A – LCA – B in the original tree.
- Therefore, total time taken will be O(logN * TimeTakenByLinearDS)

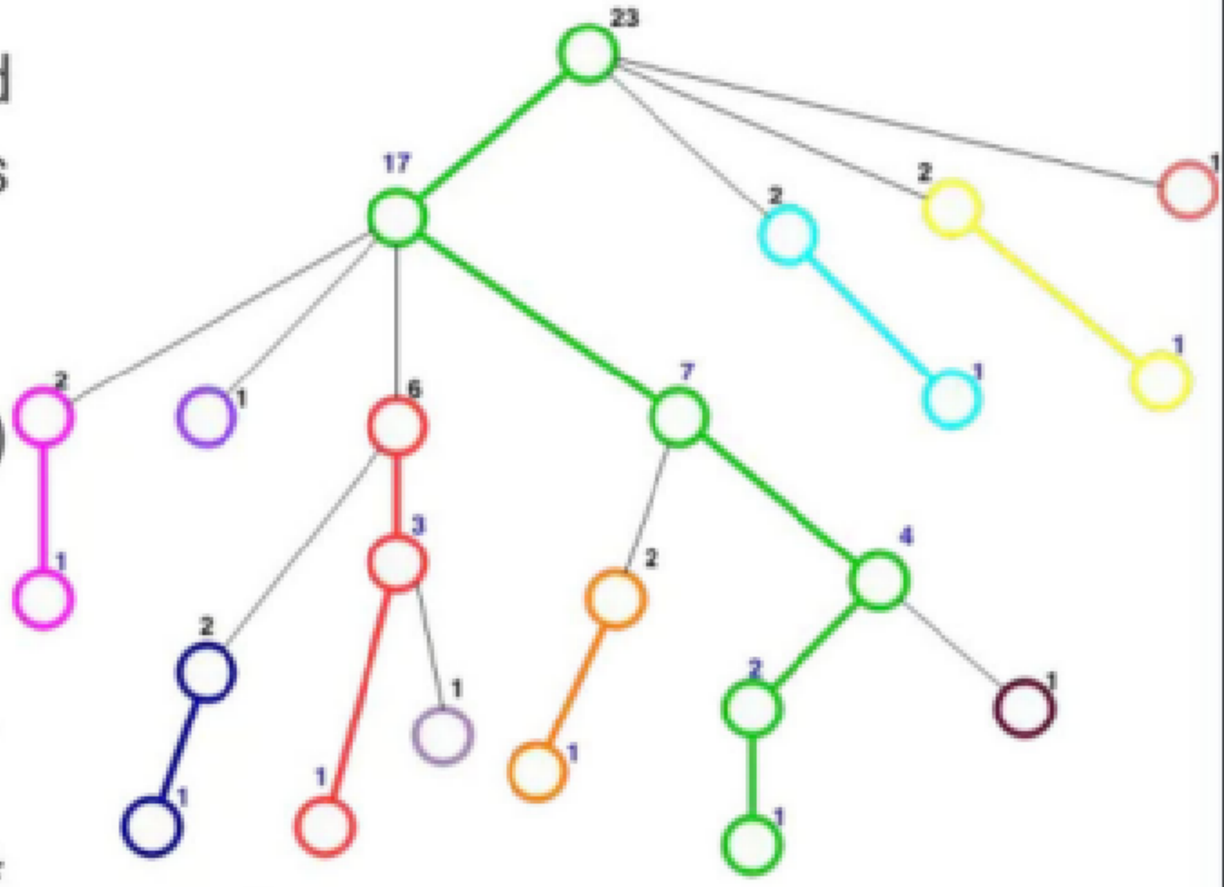


Image Source: https://blog.anudeep2011.com/heavy-light-decomposition/