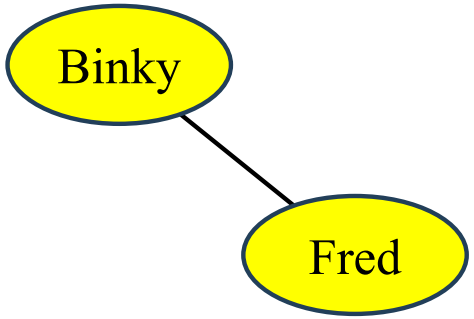
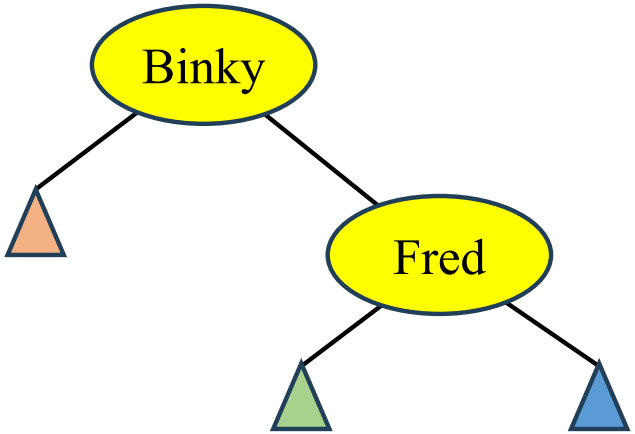
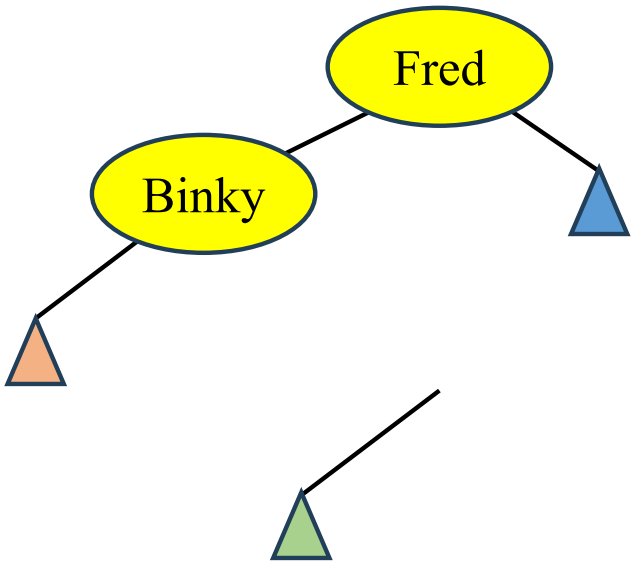


Task: Rotate about “Fred”.



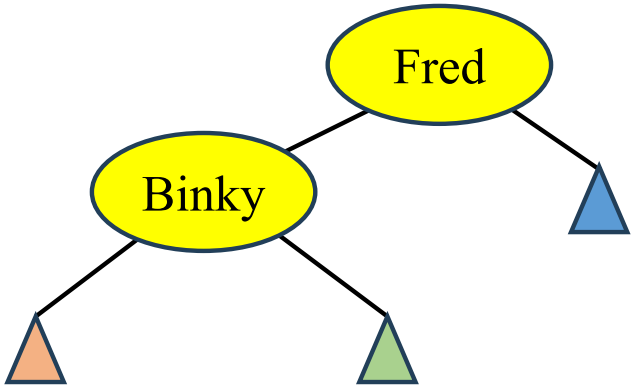
Task: Rotate about “Fred”.

1. Label the subtrees.



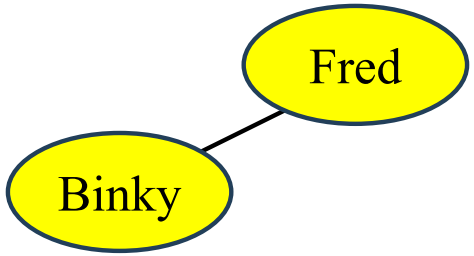
Task: Rotate about “Fred”.

1. Label the subtrees.
2. Detach the middle subtree, and make Fred the parent of Binky.



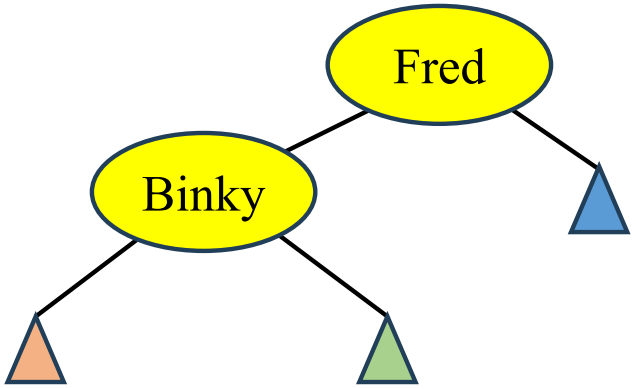
Task: Rotate about “Fred”.

1. Label the subtrees.
2. Detach the middle subtree, and make Fred the parent of Binky.
3. Connect the middle subtree.



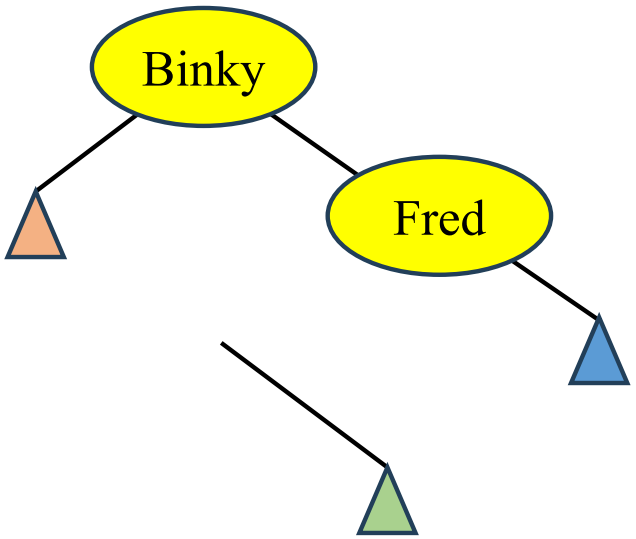
Task: Rotate about “Fred”.

1. Label the subtrees.
2. Detach the middle subtree, and make Fred the parent of Binky.
3. Connect the middle subtree.



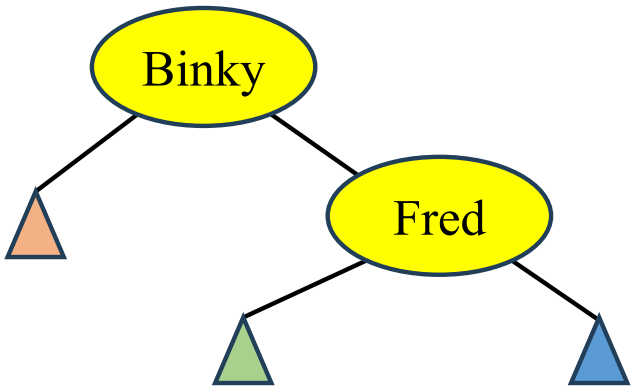
Going backward: Rotate about Binky.

1. Label the subtrees.



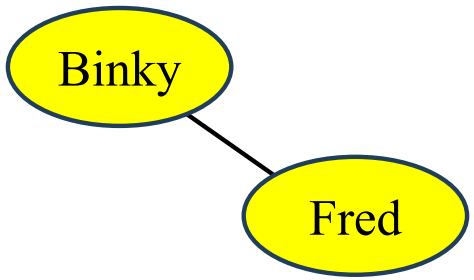
Going backward: Rotate about Binky.

1. Label the subtrees.
2. Detach the middle subtree, and make Binky the parent of Fred.

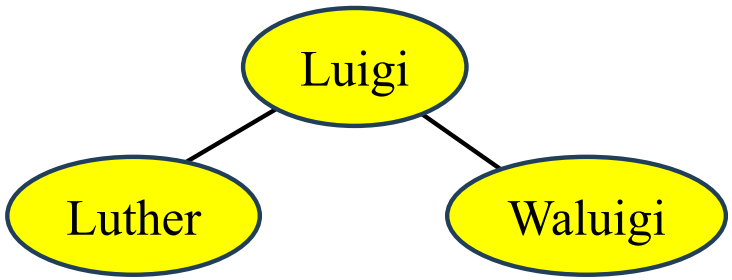


Going backward: Rotate about Binky.

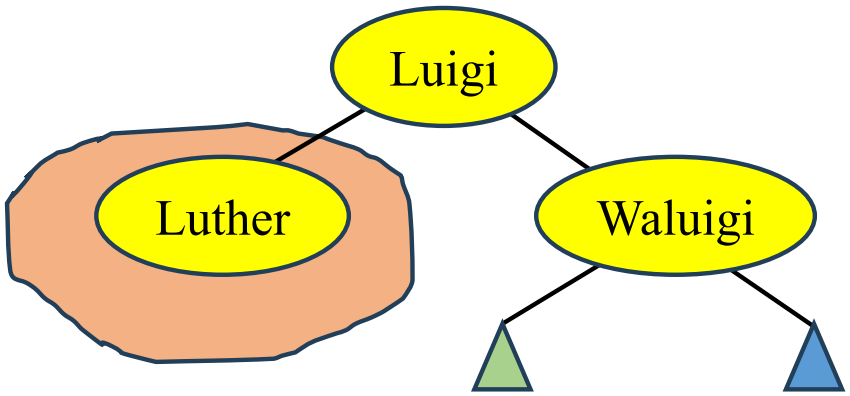
1. Label the subtrees.
2. Detach the middle subtree, and make Binky the parent of Fred.
3. Connect the middle subtree.



Going backward: Rotate about Binky.

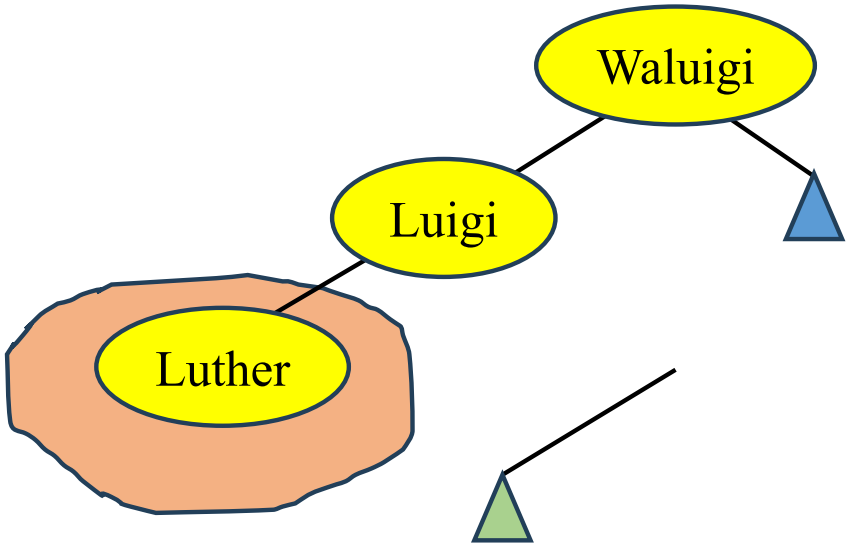


Task: Rotate about Waluigi



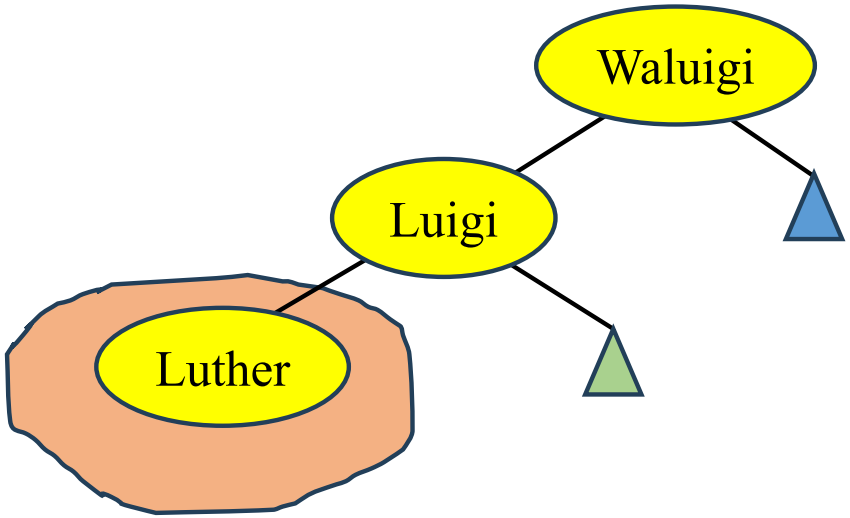
Task: Rotate about Waluigi

1. Label the subtrees.



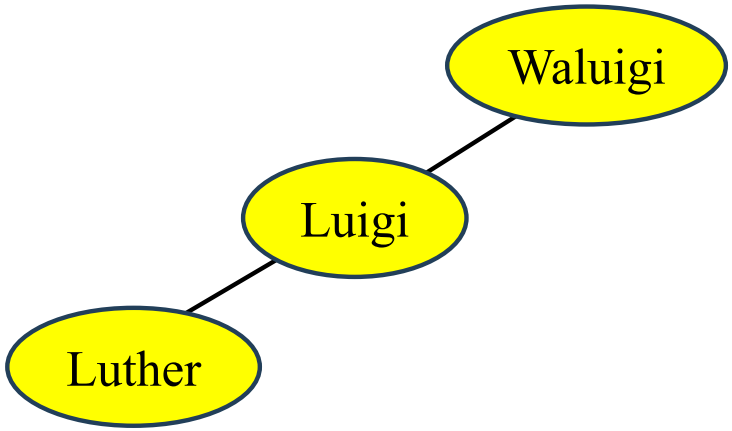
Task: Rotate about Waluigi

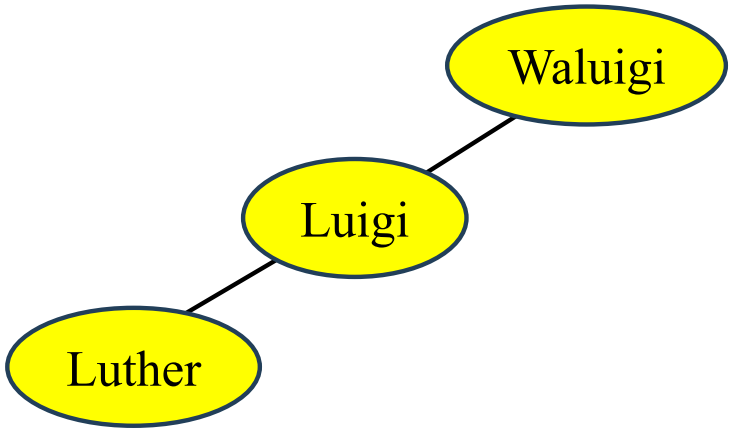
1. Label the subtrees.
2. Detach the middle subtree, and make Waluigi the parent of Luigi.



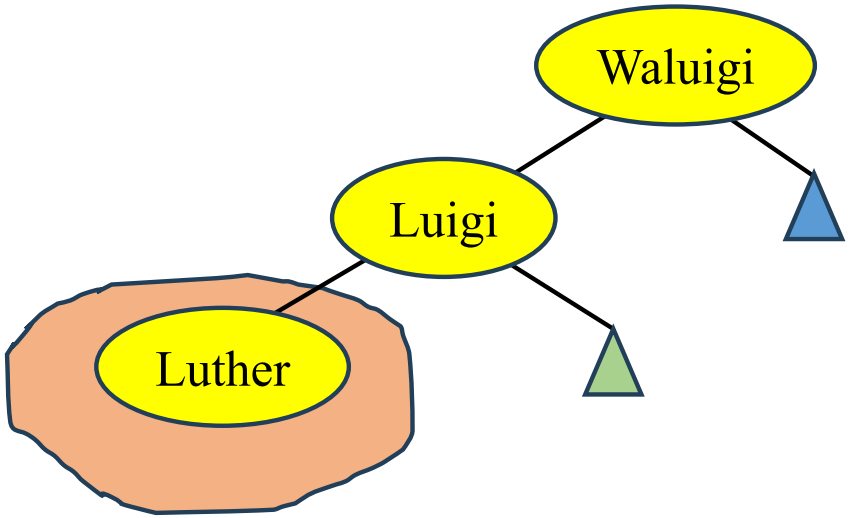
Task: Rotate about Waluigi

1. Label the subtrees.
2. Detach the middle subtree, and make Waluigi the parent of Luigi.
3. Reattach the subtree.


Waluigi
Luigi
Luther

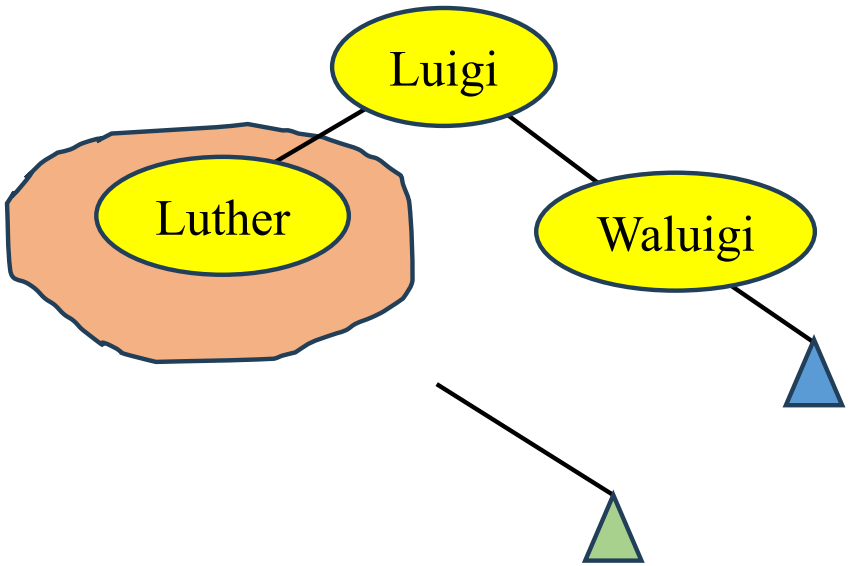


Task: Rotate about Luigi



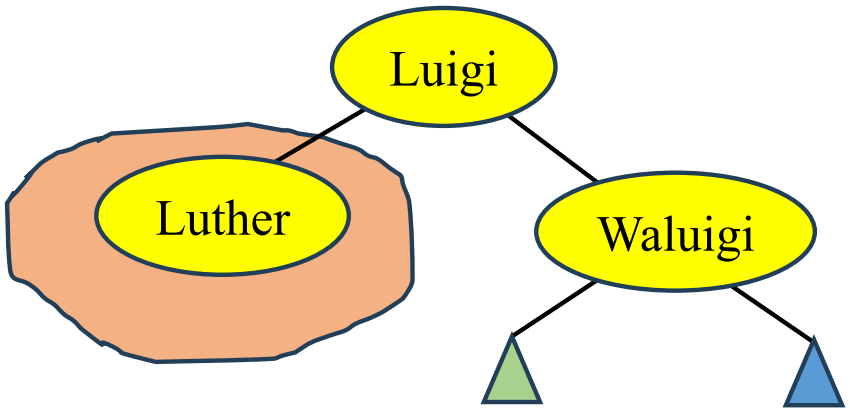
Task: Rotate about Luigi

1. Label the subtrees.



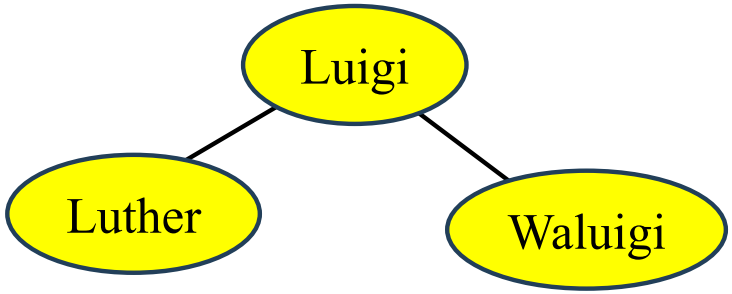
Task: Rotate about Luigi

1. Label the subtrees.
2. Detach the middle subtree, and make Luigi the parent of Waluigi.

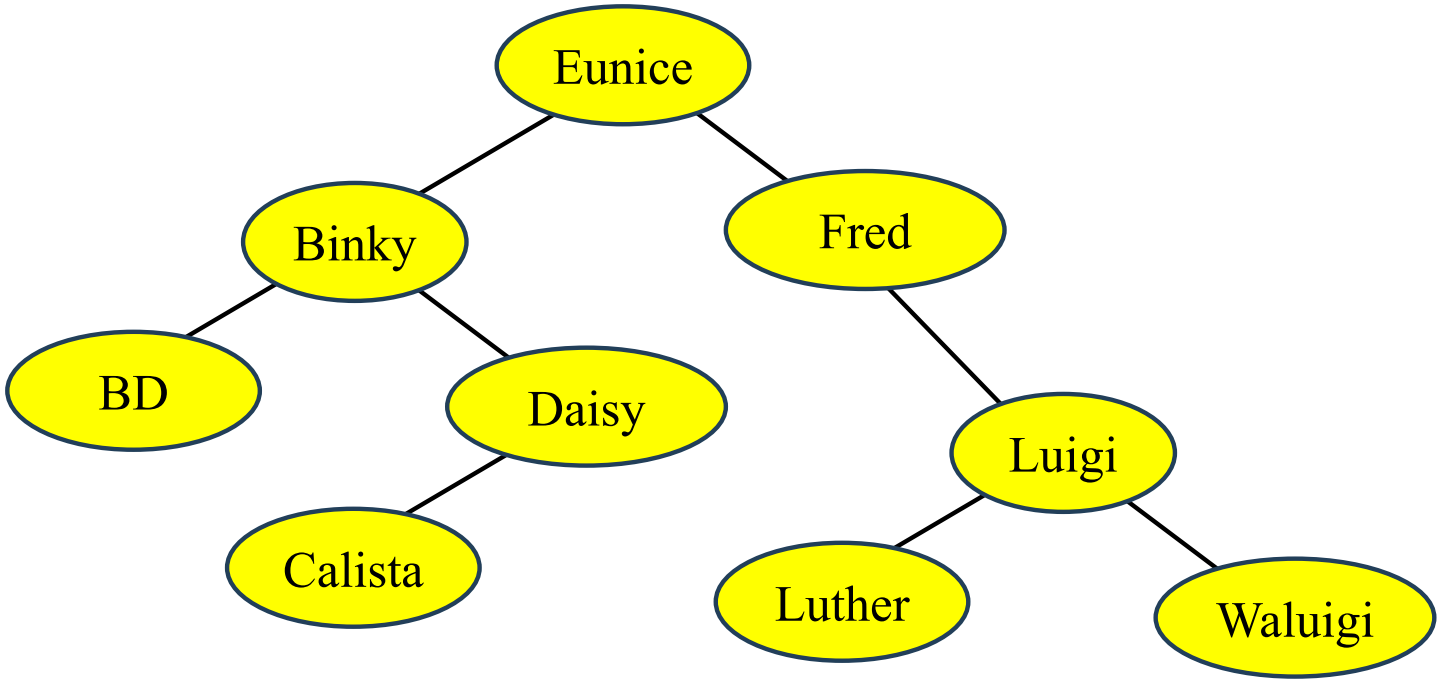


Task: Rotate about Luigi

1. Label the subtrees.
2. Detach the middle subtree, and make Luigi the parent of Waluigi.
3. Reattach the subtree.

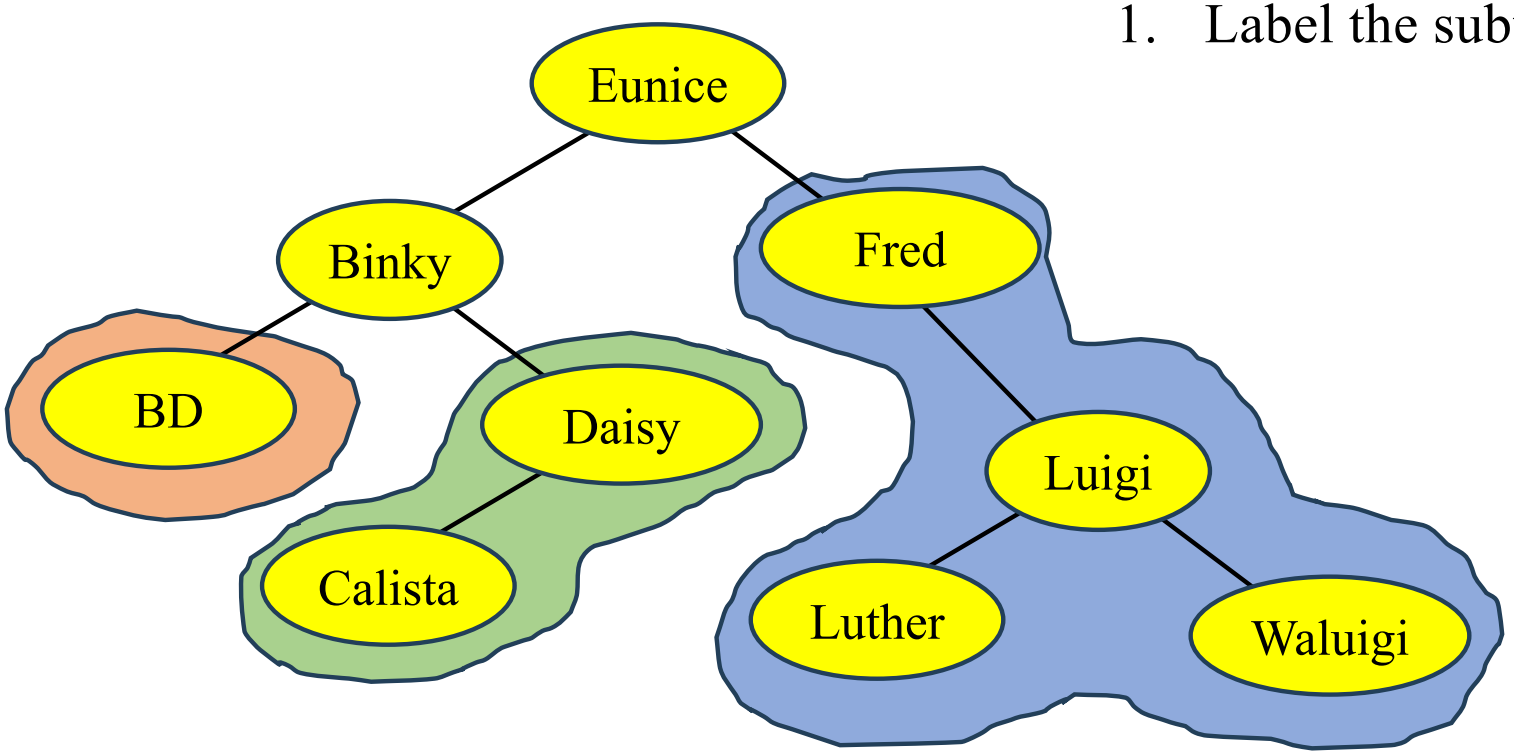

Luigi
Luther
Waluigi

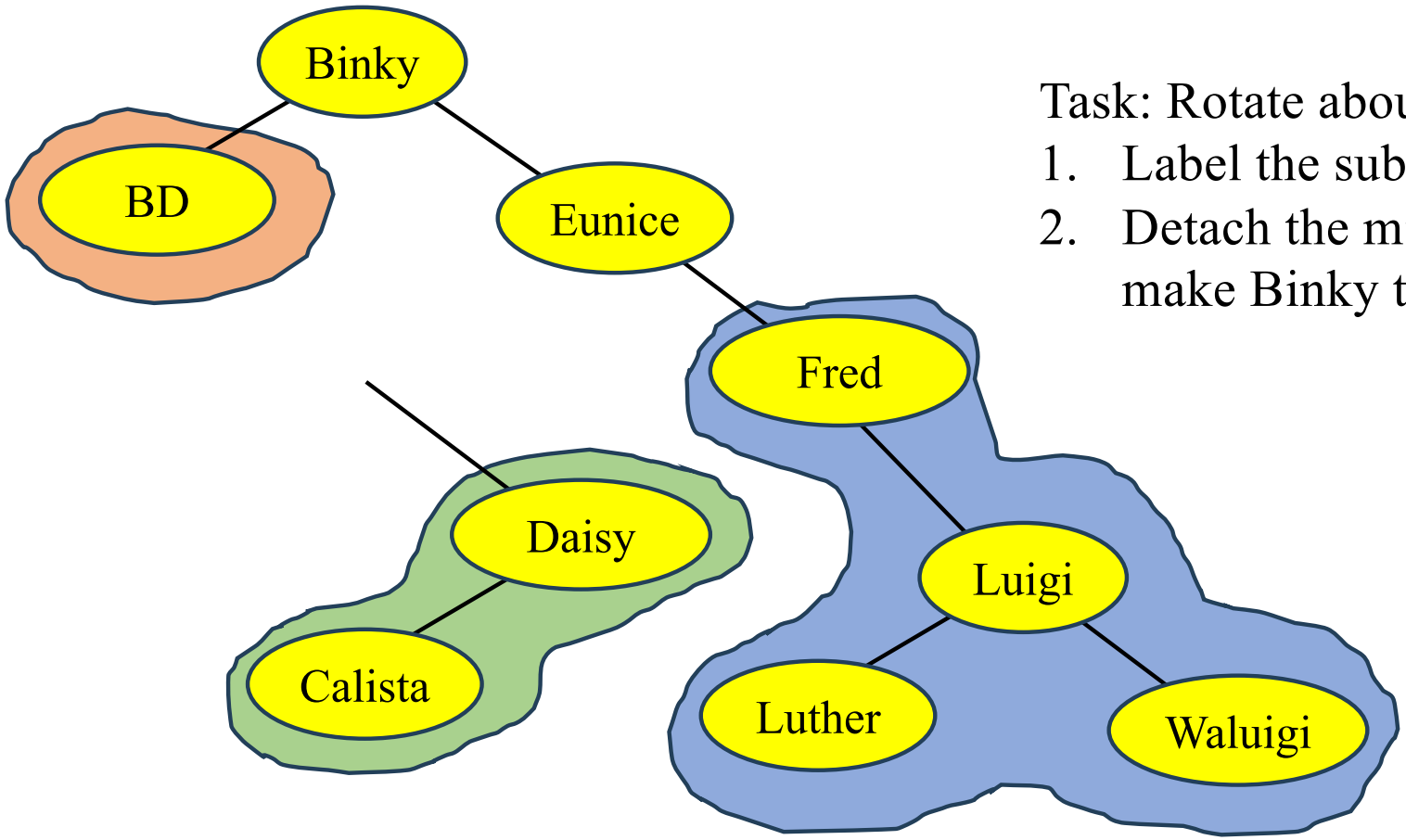
Task: Rotate about Binky



Task: Rotate about Binky

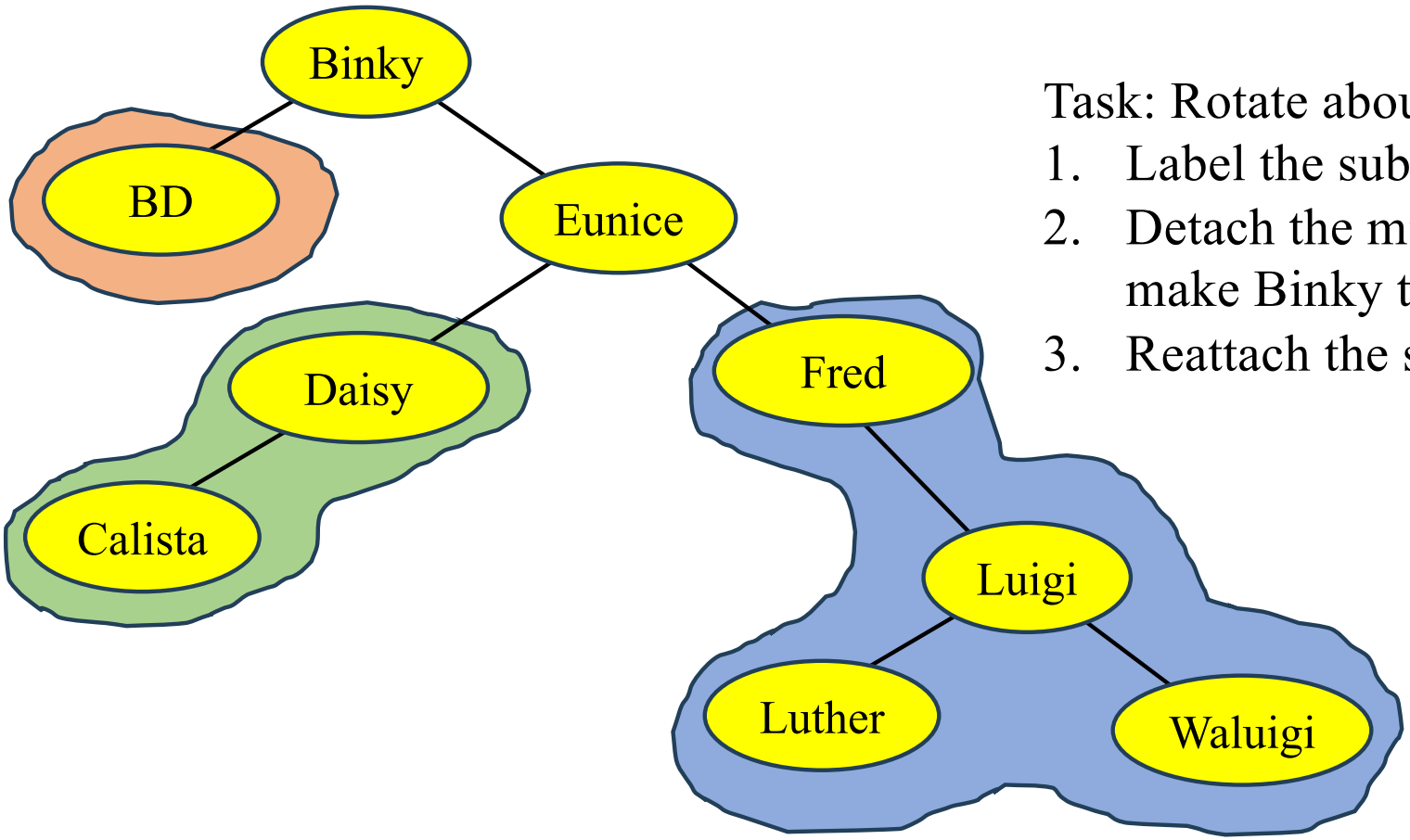
1. Label the subtrees.





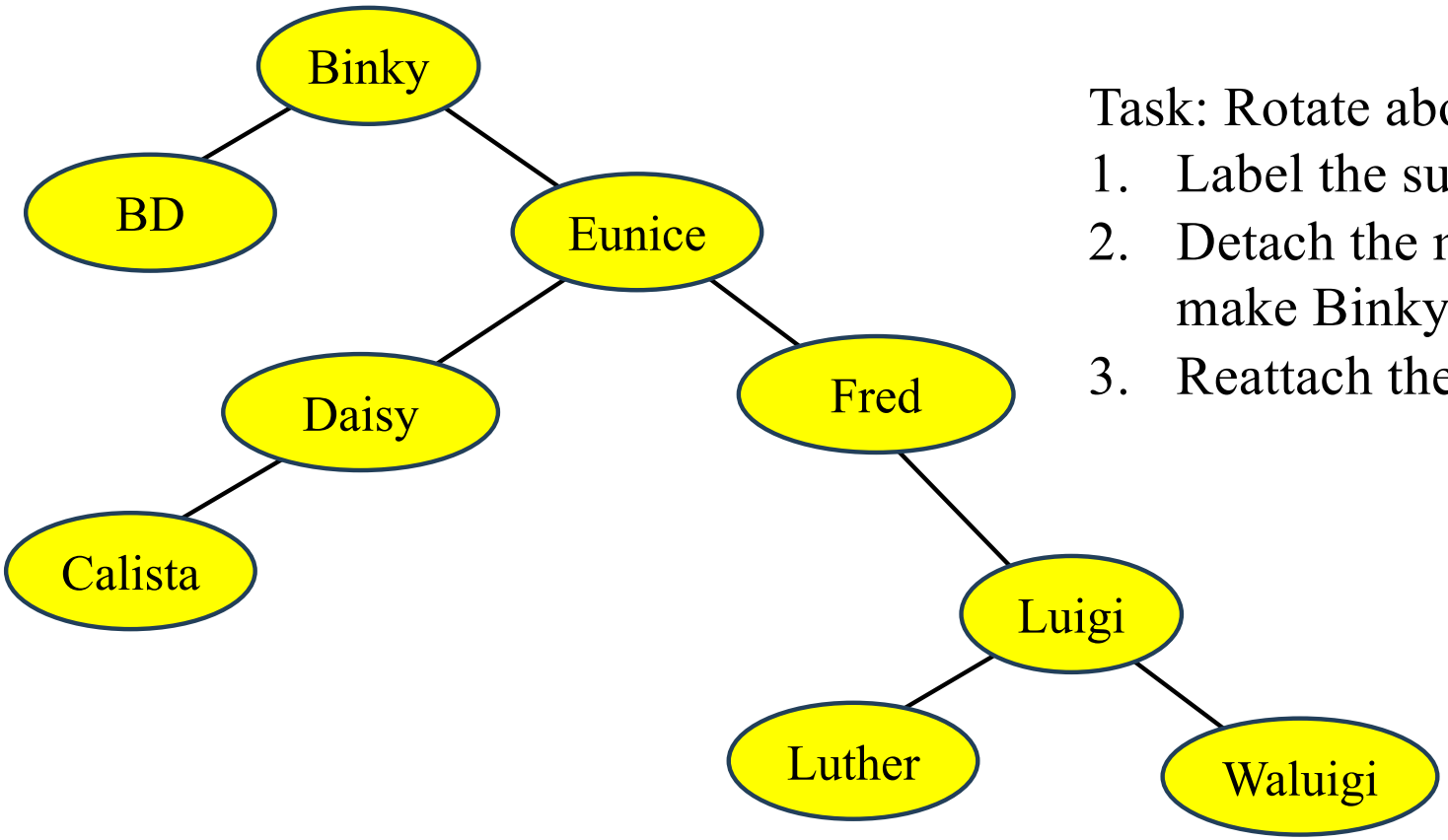
Task: Rotate about Binky

1. Label the subtrees.
2. Detach the middle subtree, and make Binky the parent of Eunice.



Task: Rotate about Binky

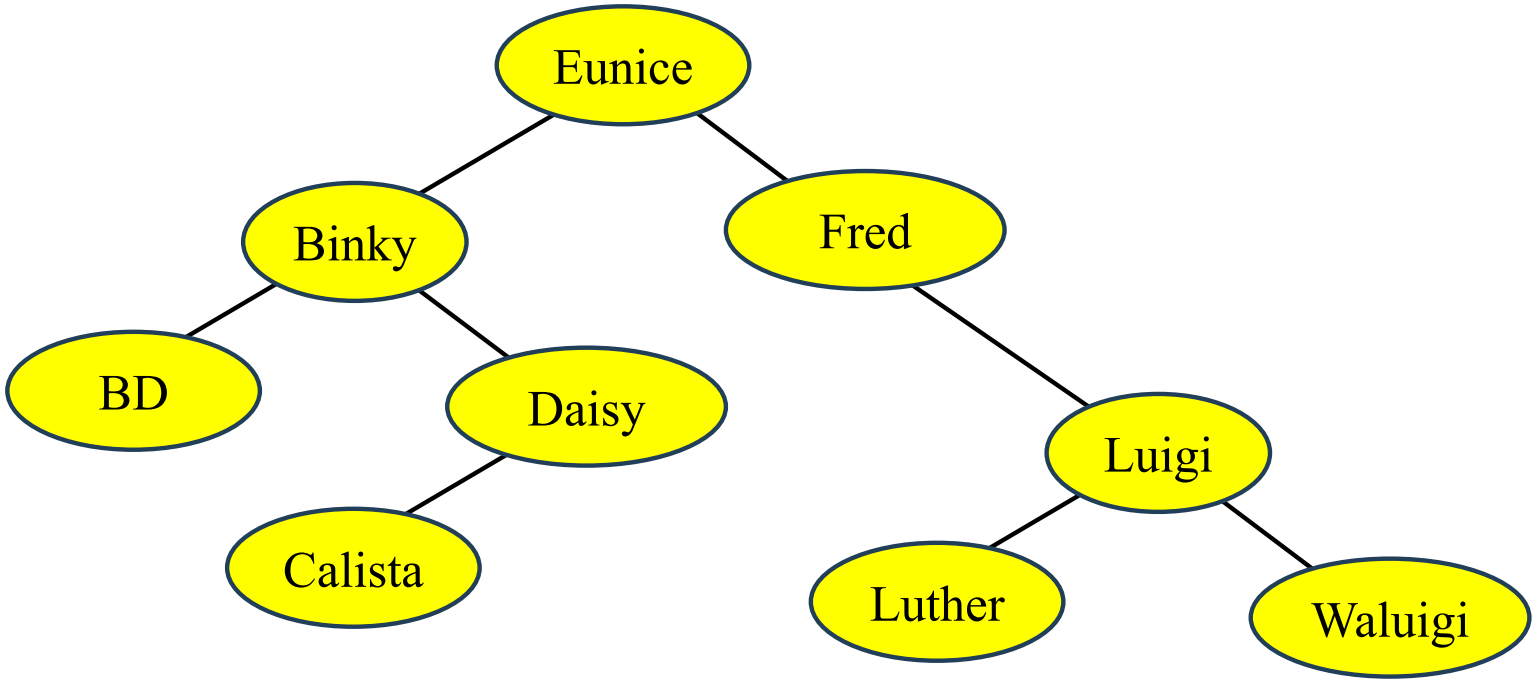
1. Label the subtrees.
2. Detach the middle subtree, and make Binky the parent of Eunice.
3. Reattach the subtree.



Task: Rotate about Binky

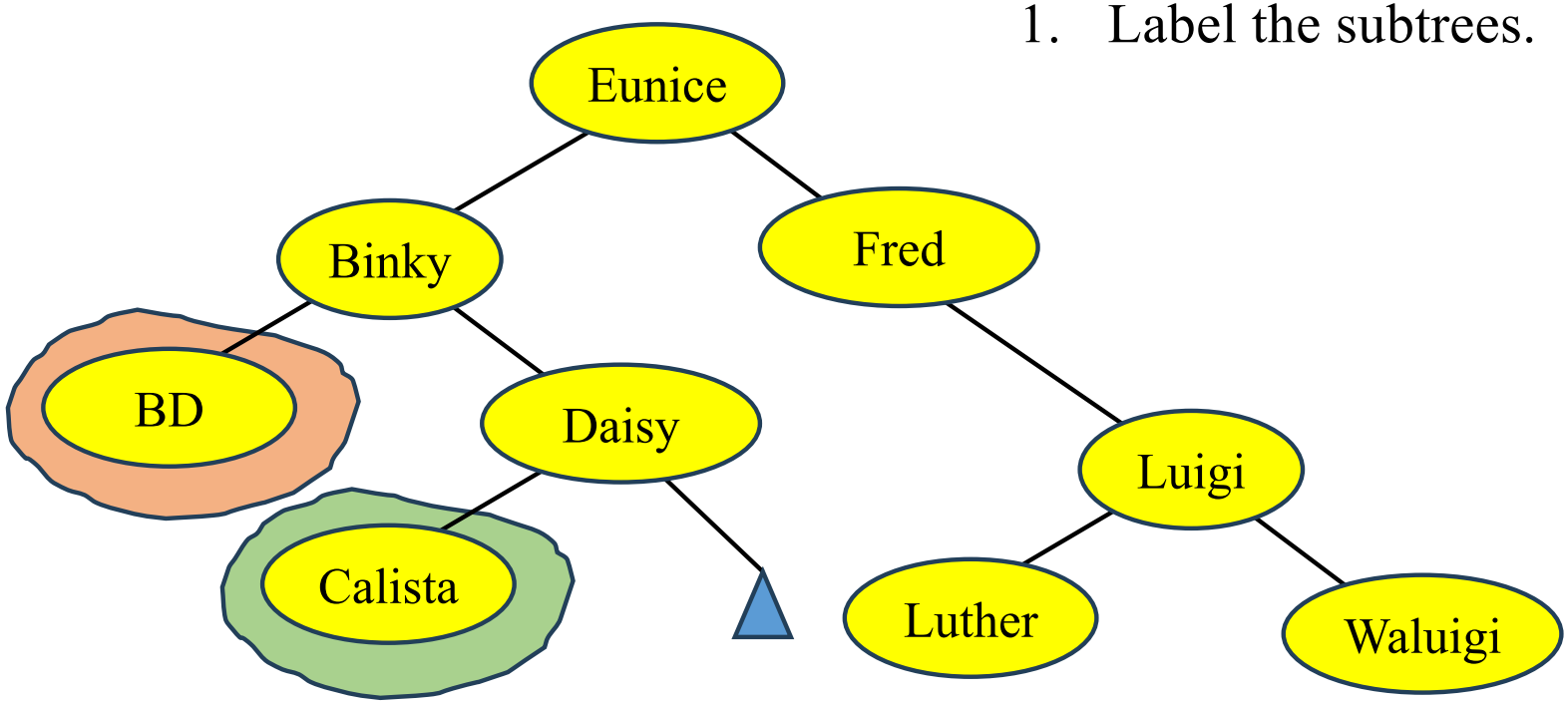
1. Label the subtrees.
2. Detach the middle subtree, and make Binky the parent of Eunice.
3. Reattach the subtree.

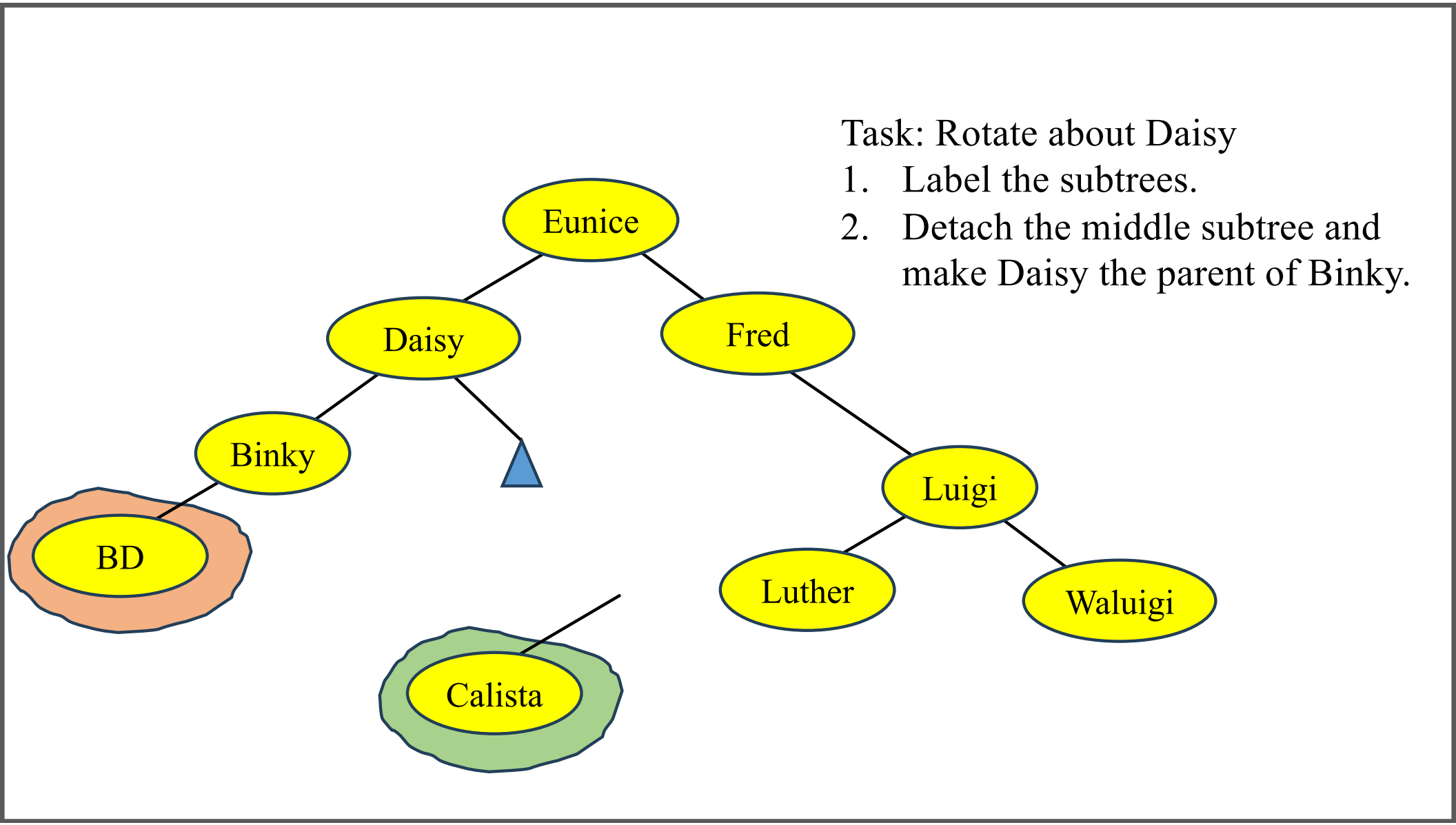
Task: Rotate about Daisy

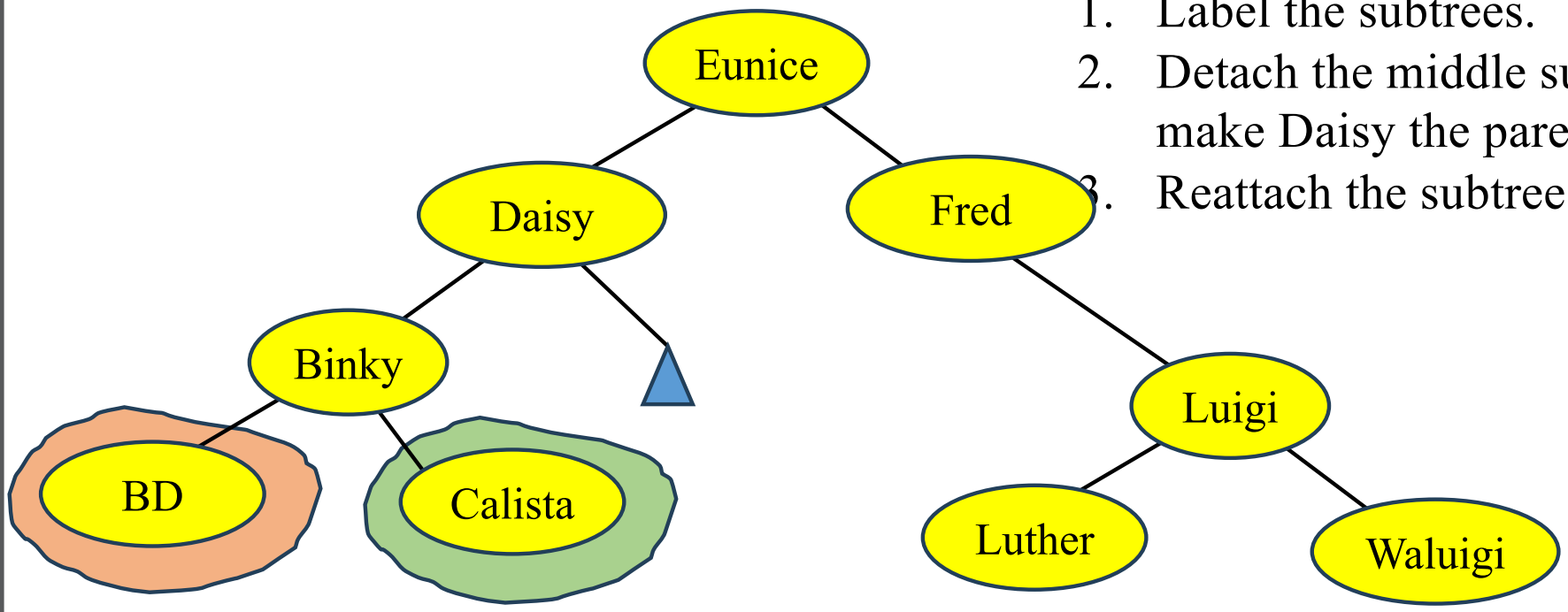


Task: Rotate about Daisy

1. Label the subtrees.

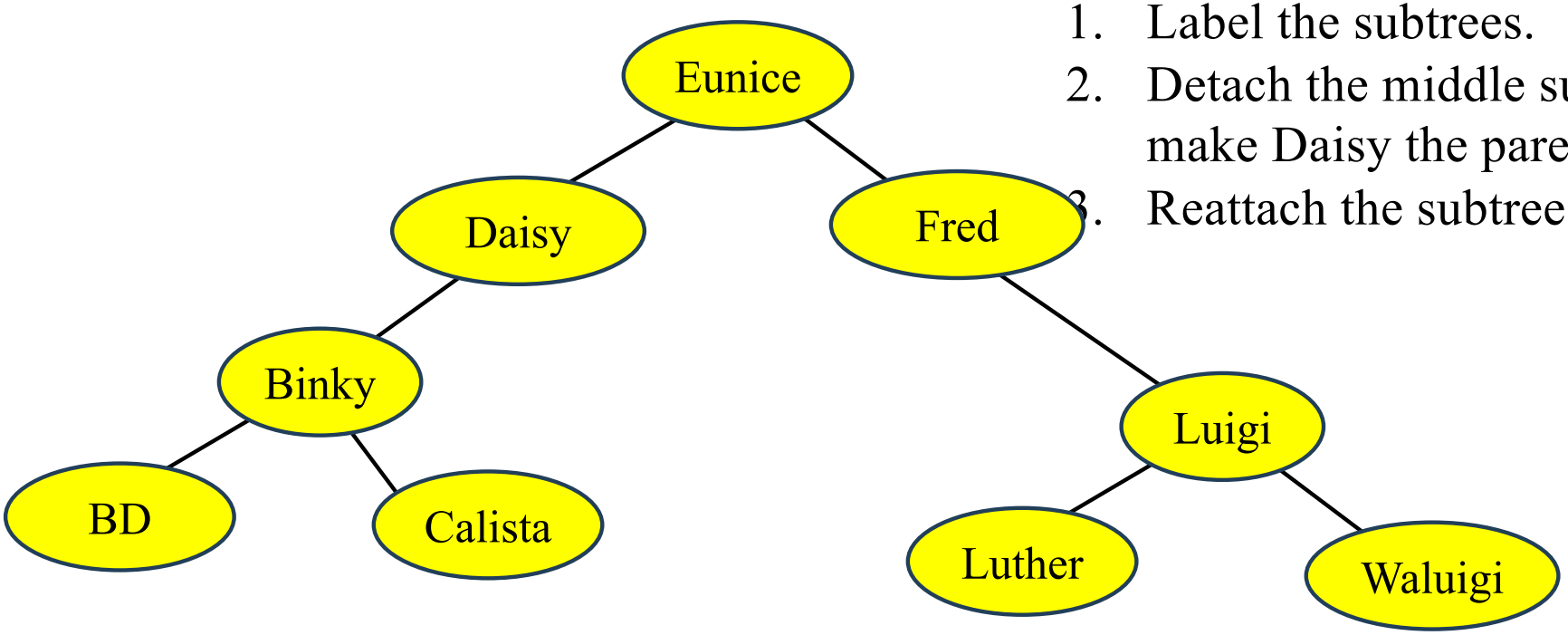



Task: Rotate about Daisy
1. Label the subtrees.
2. Detach the middle subtree and make Daisy the parent of Binky.
Eunice
Daisy
Fred
Binky
Luigi
BD
Luther
Waluigi
Calista



Task: Rotate about Daisy

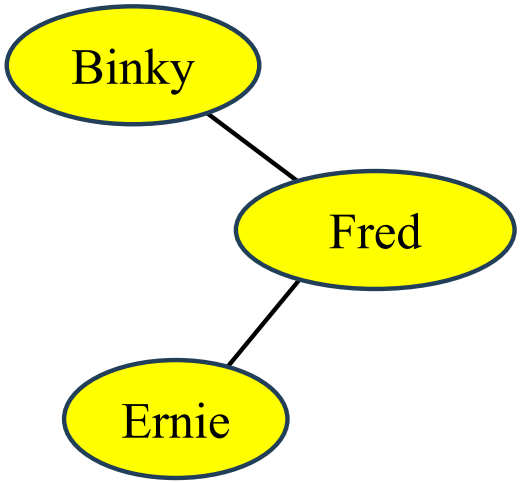
1. Label the subtrees.
2. Detach the middle subtree and make Daisy the parent of Binky.
3. Reattach the subtree.

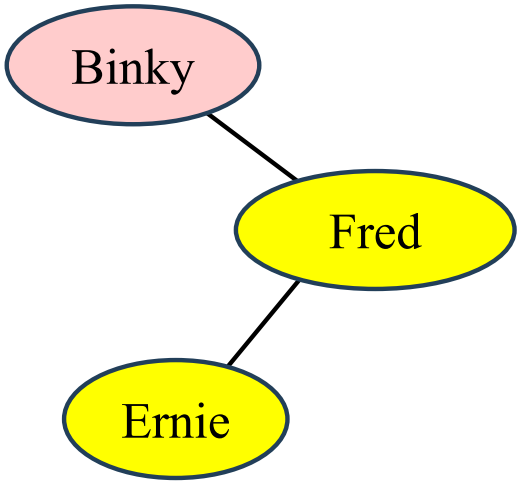


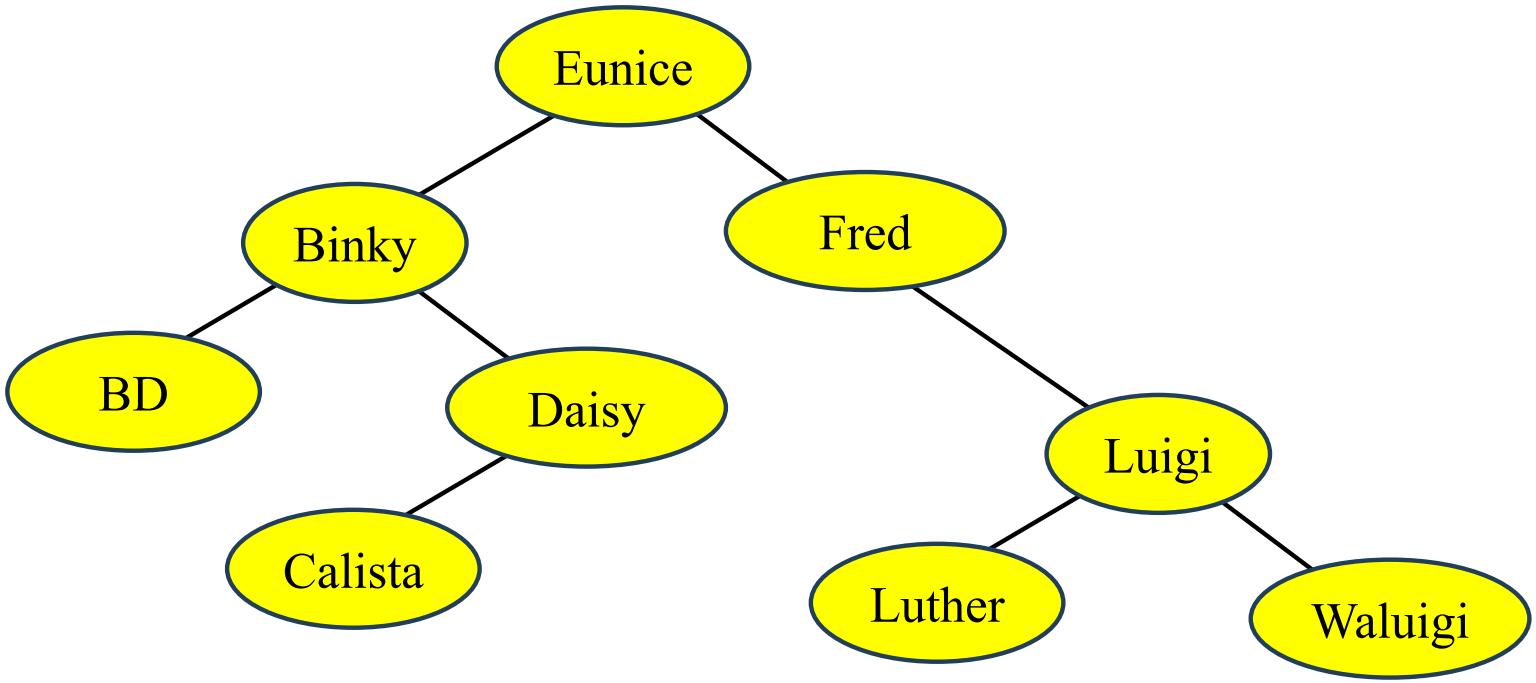
Task: Rotate about Daisy

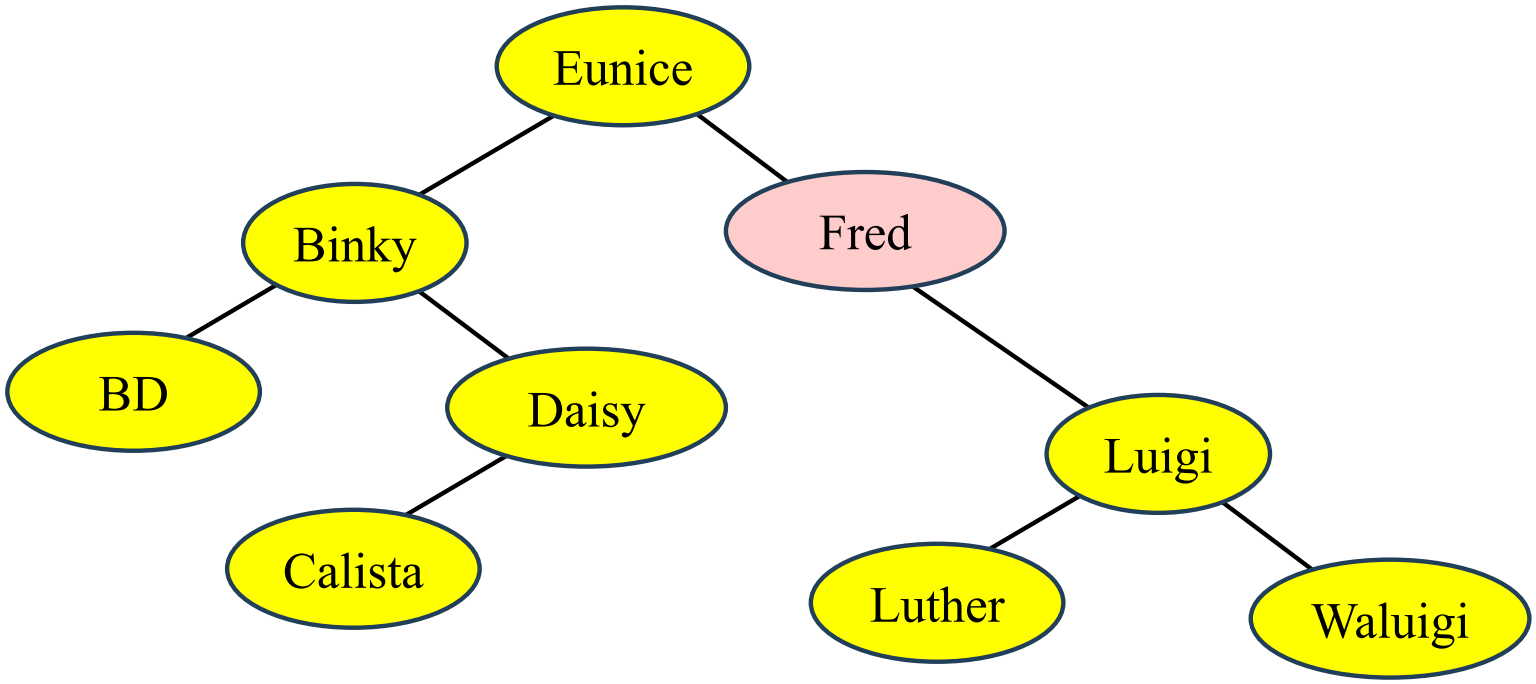
1. Label the subtrees.
2. Detach the middle subtree and make Daisy the parent of Binky.
3. Reattach the subtree.

Is this an AVL Tree?

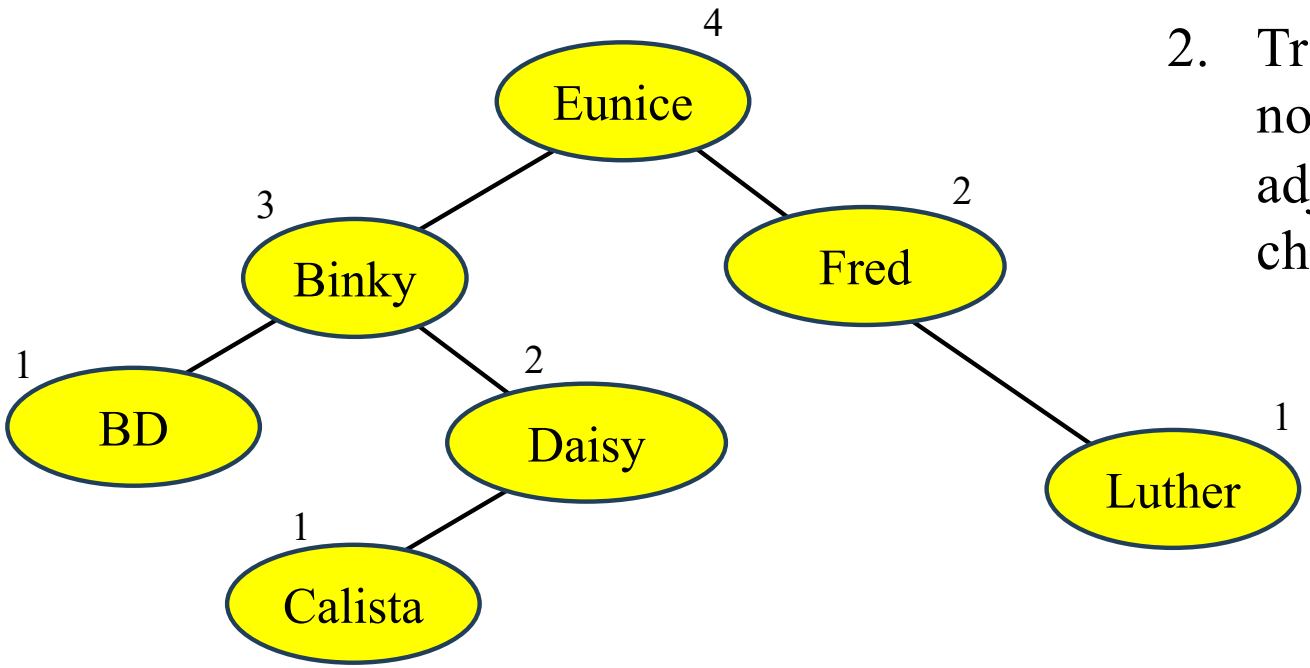

Binky
Fred
Ernie


Binky
Fred
Ernie


Eunice
Binky
Fred
BD
Daisy
Luigi
Calista
Luther
Waluigi


Eunice
Binky
Fred
BD
Daisy
Luigi
Calista
Luther
Waluigi

Inserting into an AVL Tree

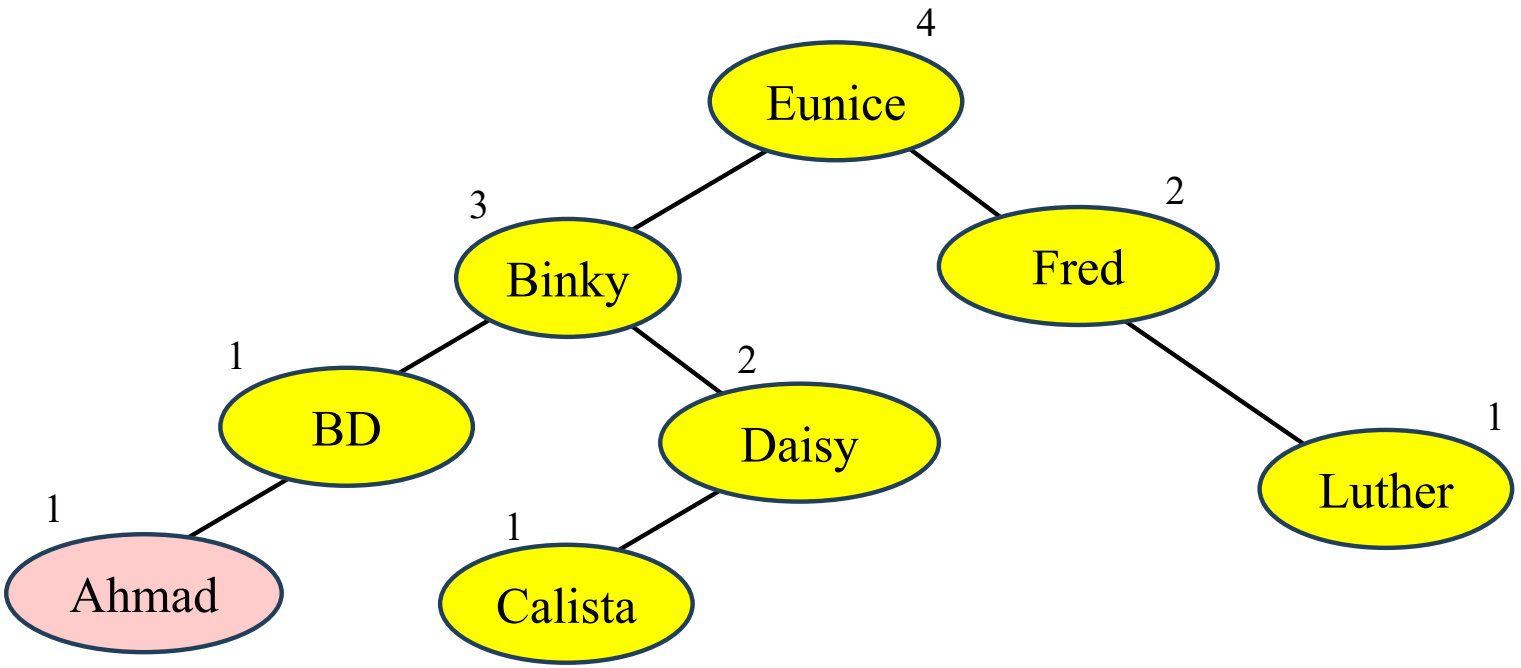


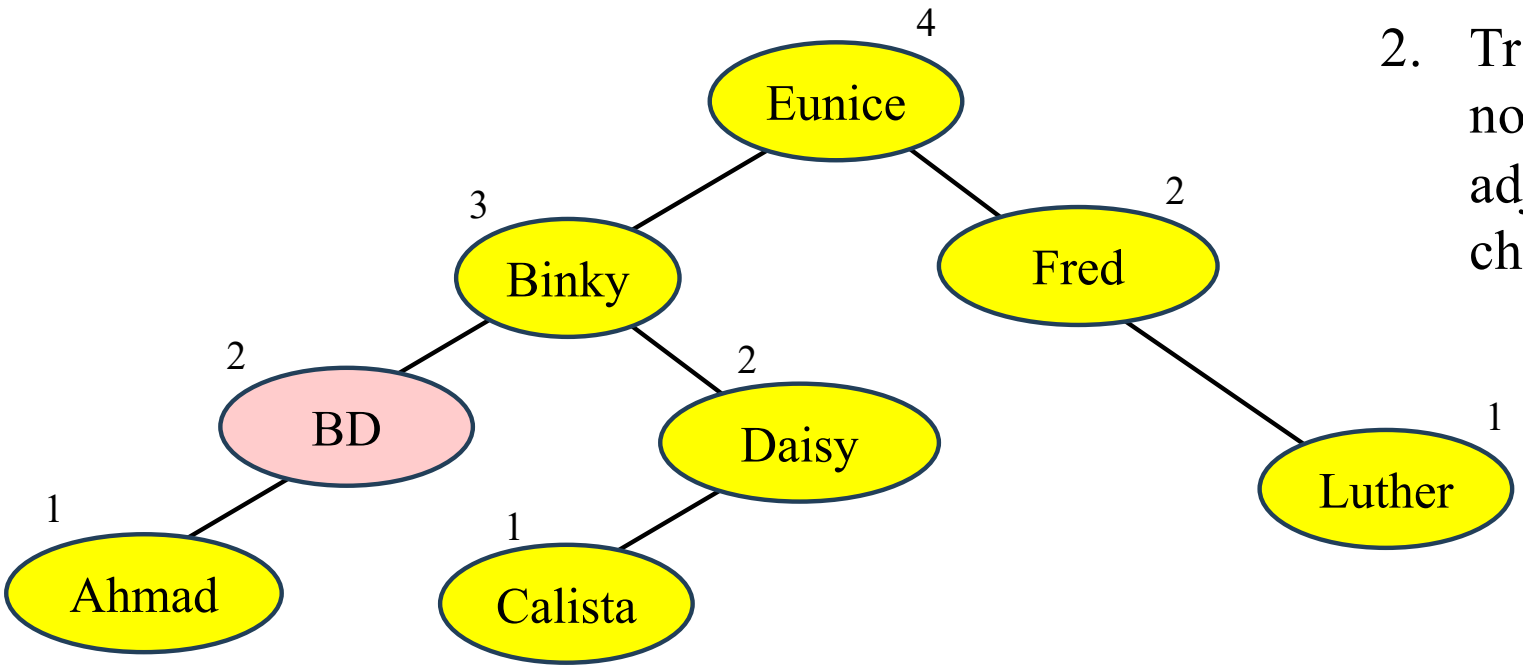
Task: Insert Ahmad.

1. Insert the node.
2. Travel from the new node to the root, adjusting heights and checking for imbalances.

Task: Insert Ahmad.

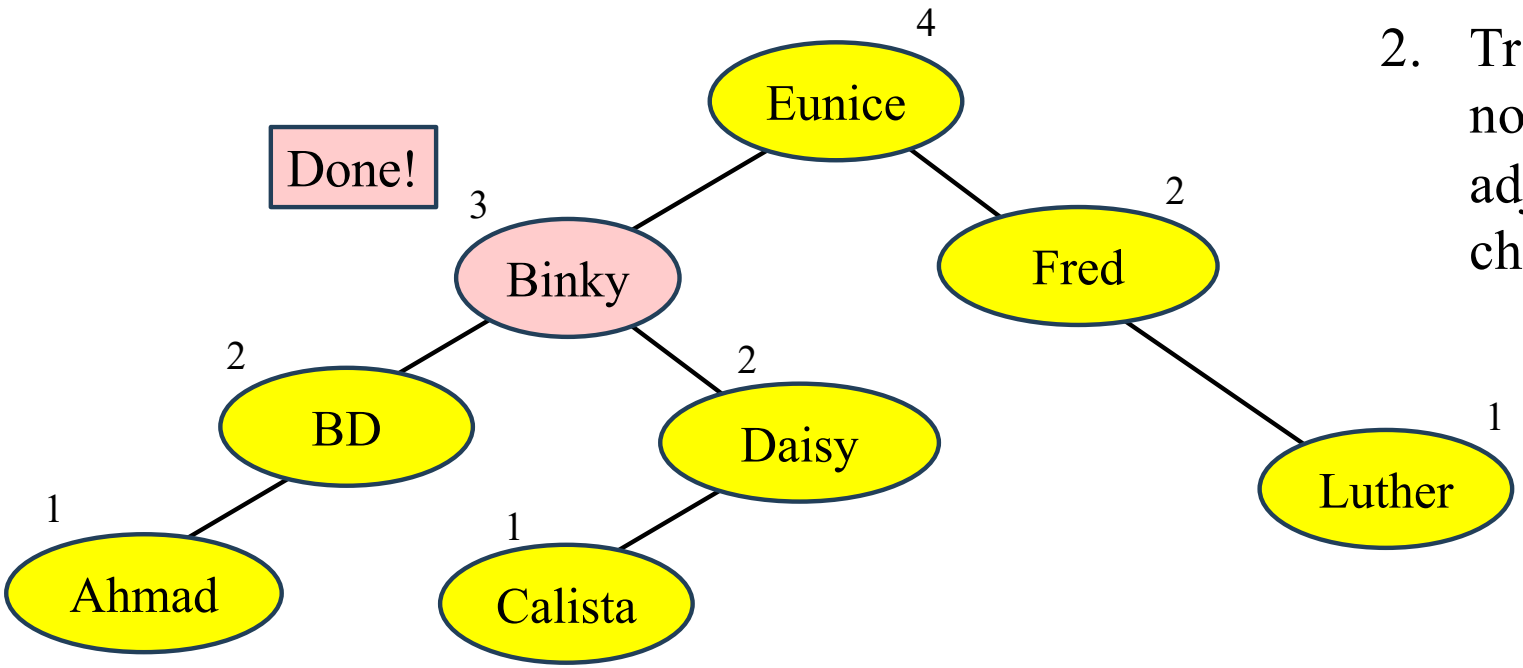
1. Insert the node.





Task: Insert Ahmad.

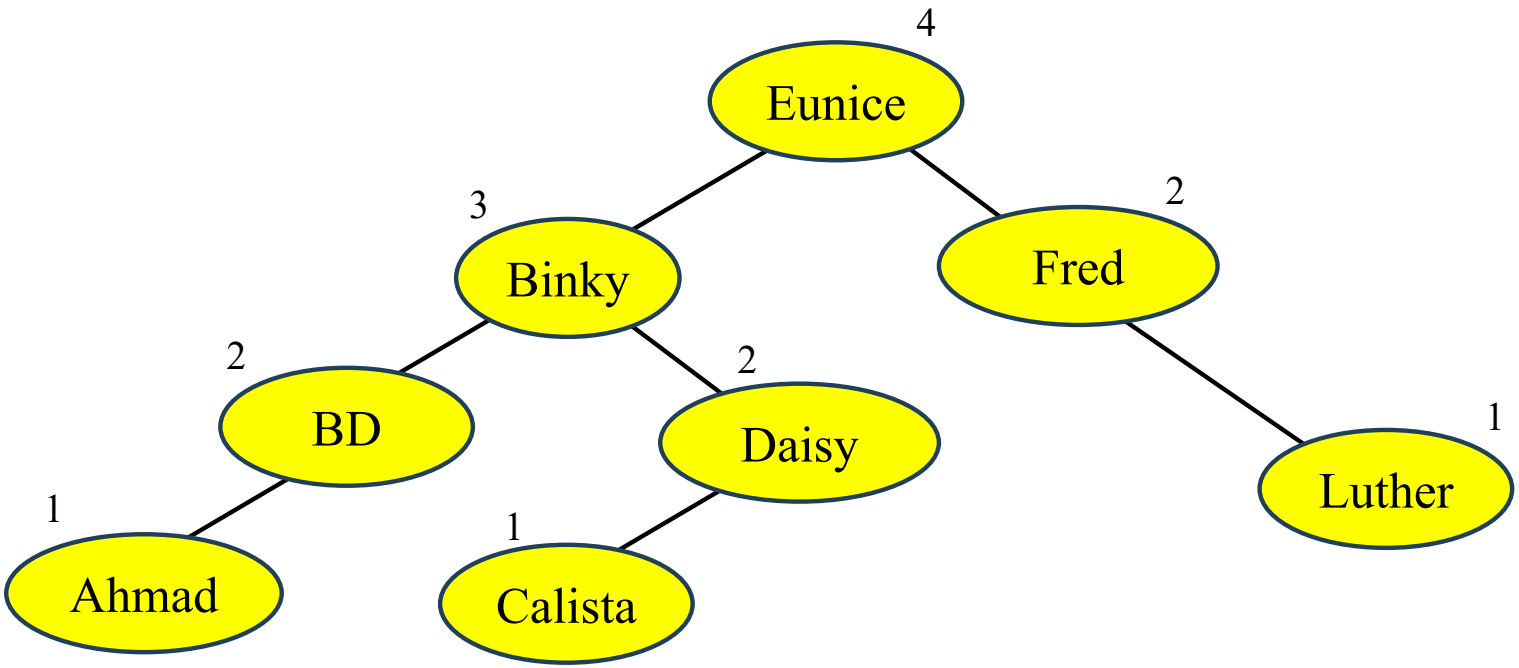
1. Insert the node.
2. Travel from the new node to the root, adjusting heights and checking for imbalances.



Task: Insert Ahmad.

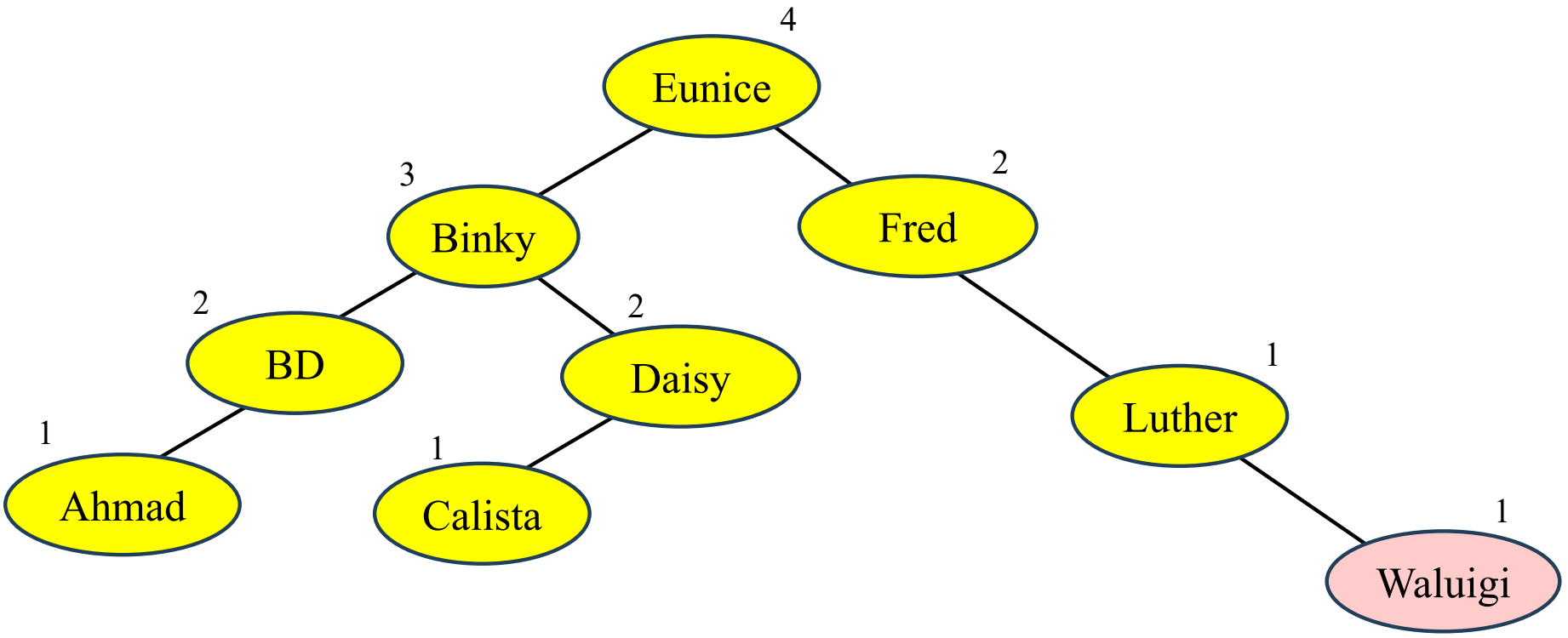
1. Insert the node.
2. Travel from the new node to the root, adjusting heights and checking for imbalances.

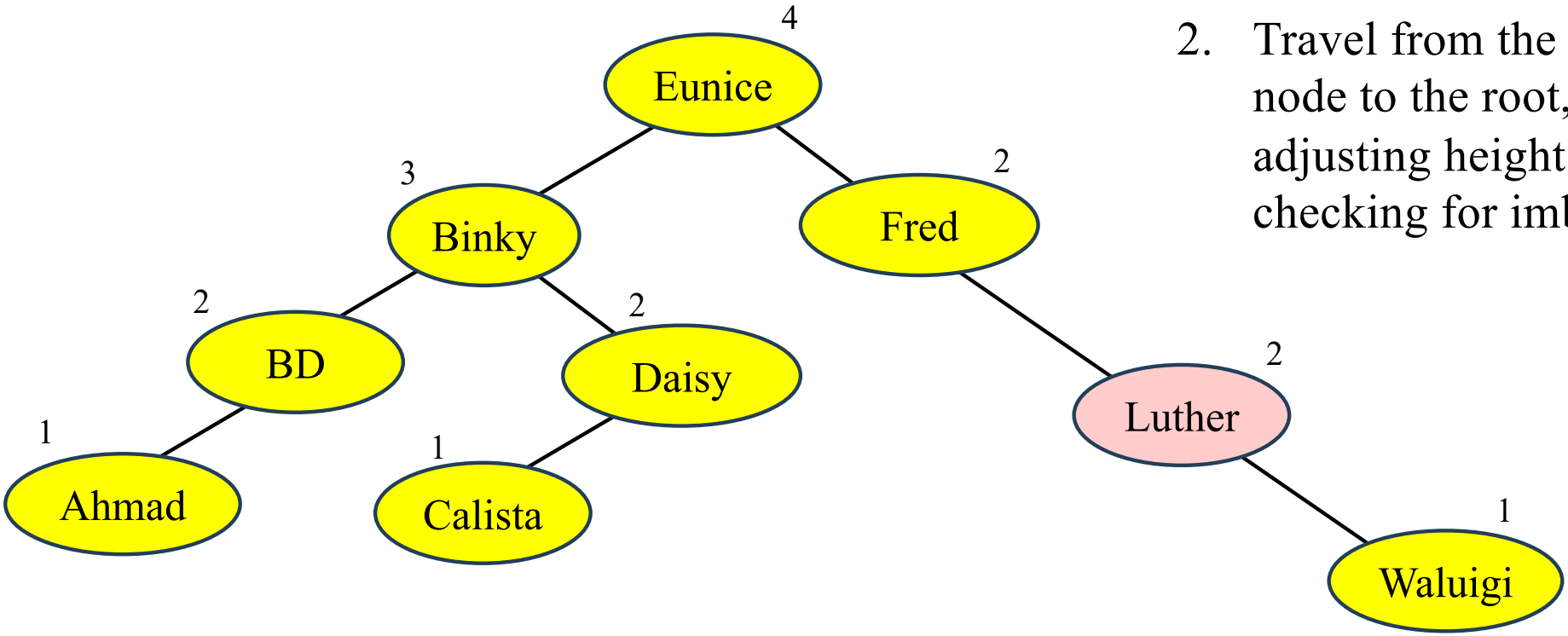
Task: Insert Waluigi.



Task: Insert Waluigi.

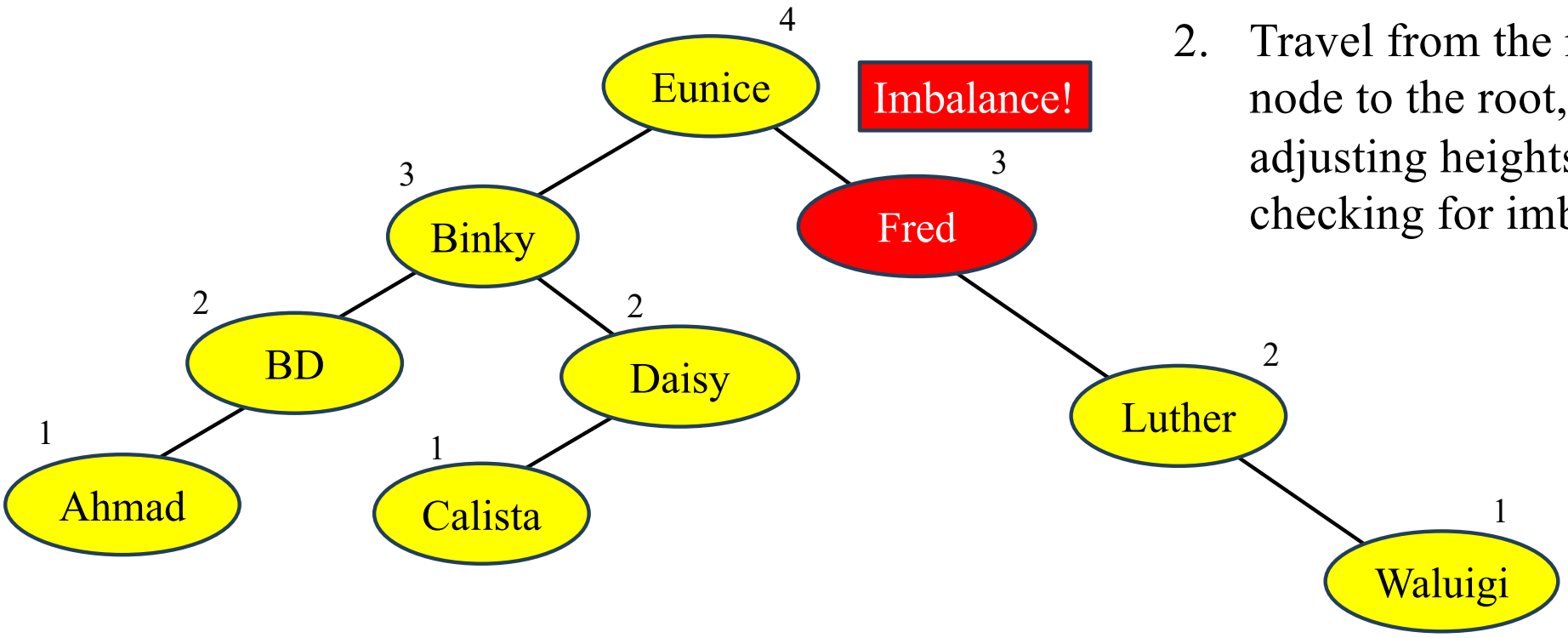
1. Insert the node.





Task: Insert Waluigi.

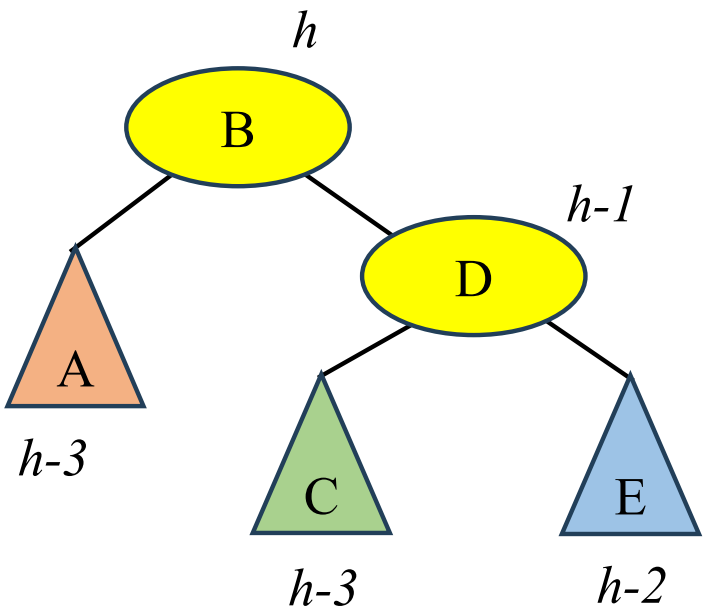
1. Insert the node.
2. Travel from the new node to the root, adjusting heights and checking for imbalances.



Task: Insert Waluigi.

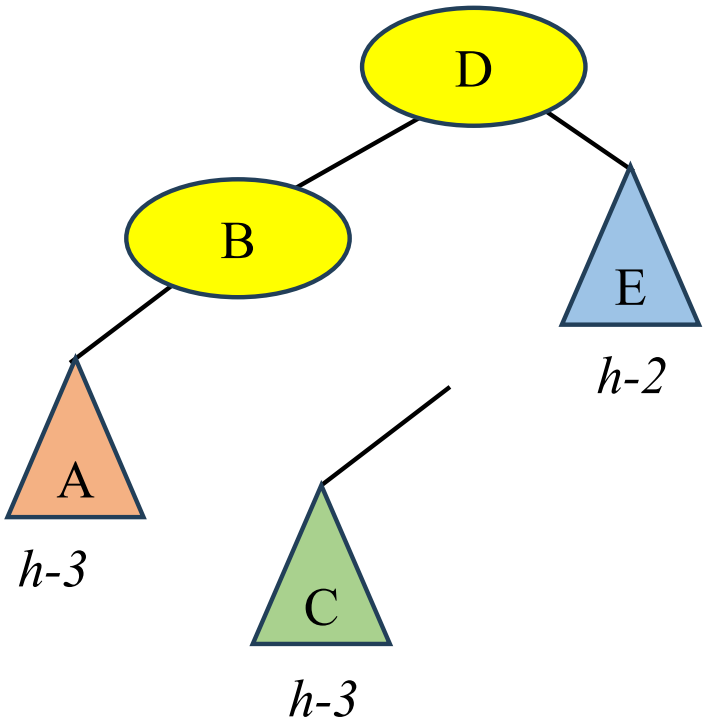
1. Insert the node.
2. Travel from the new node to the root, adjusting heights and checking for imbalances.

Fixing Zig-Zig



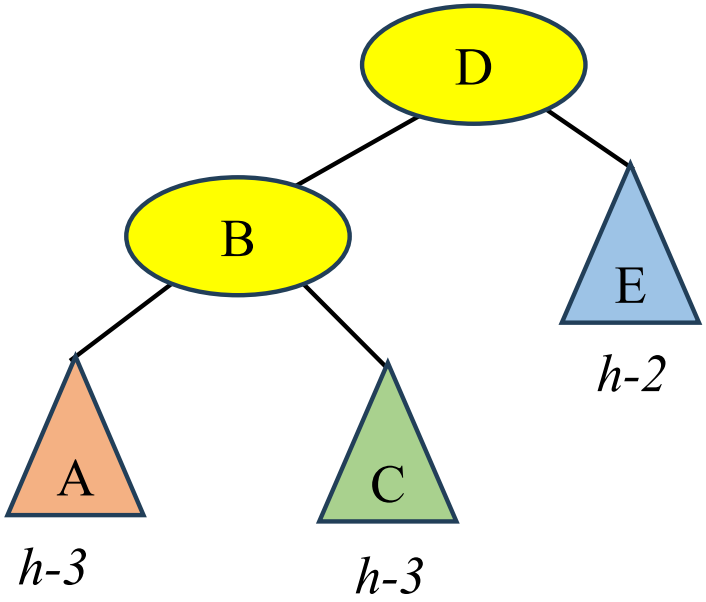
Task: Zig-Zig Rebalance

1. Rotate about D.
2. Detach the middle tree and make D the parent of B.
3. Switch C to be the child of B.



Task: Zig-Zig Rebalance

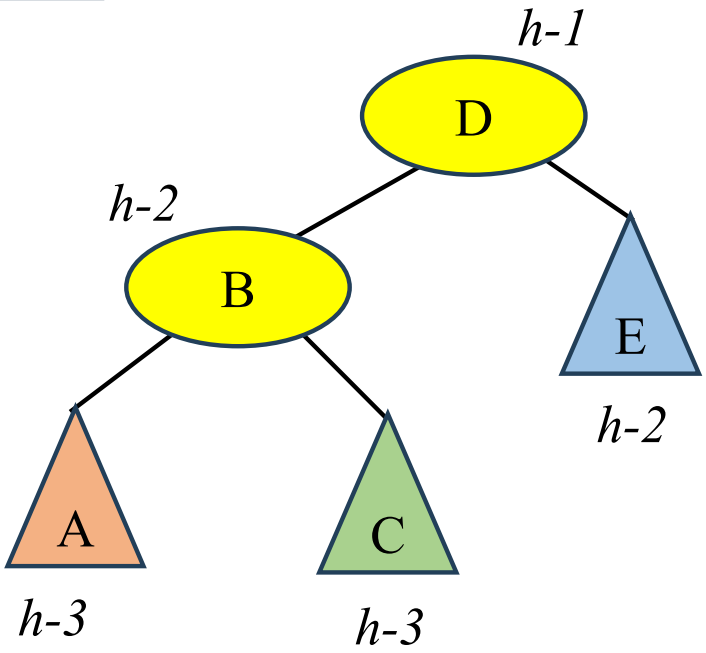
1. Rotate about D.
2. Detach the middle tree and make D the parent of B.



Task: Zig-Zig Rebalance

1. Rotate about D.
2. Detach the middle tree and make D the parent of B.
3. Switch C to be the child of B.

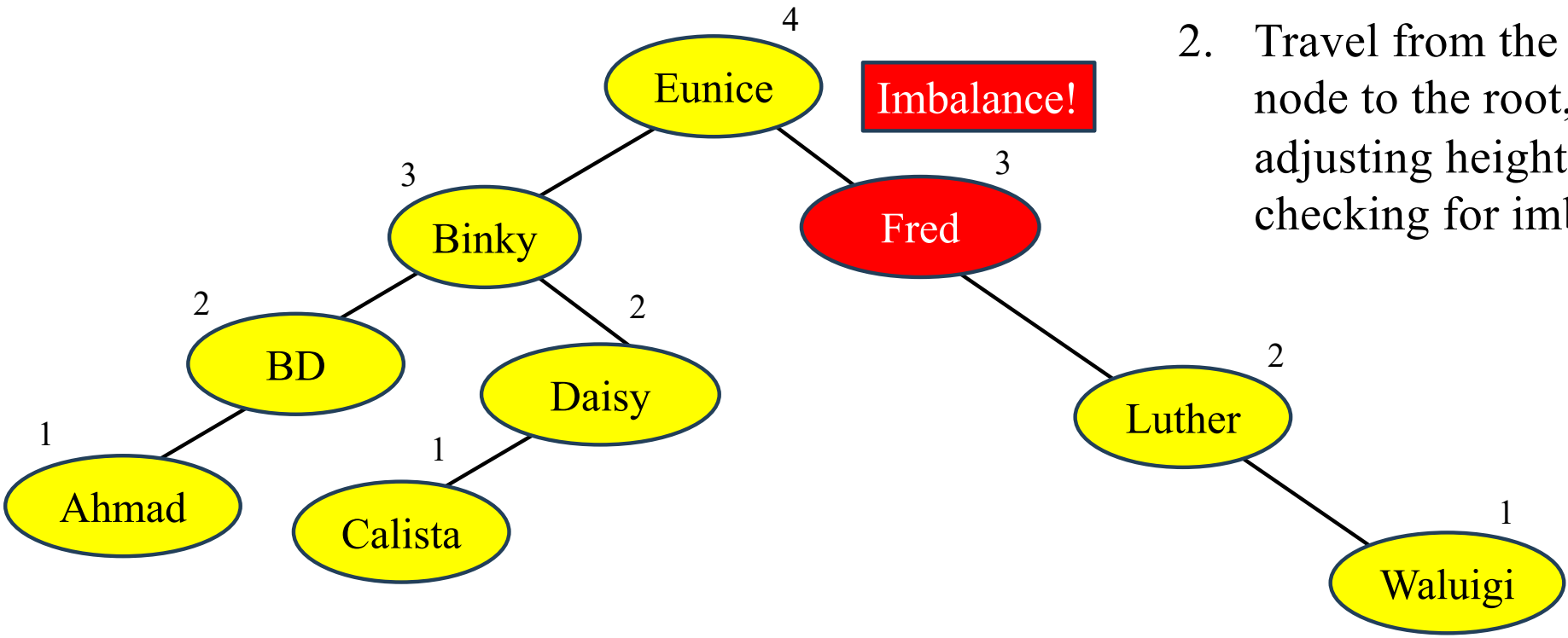
Done!



Task: Zig-Zig Rebalance

1. Rotate about D.
2. Detach the middle tree and make D the parent of B.
3. Switch C to be the child of B.

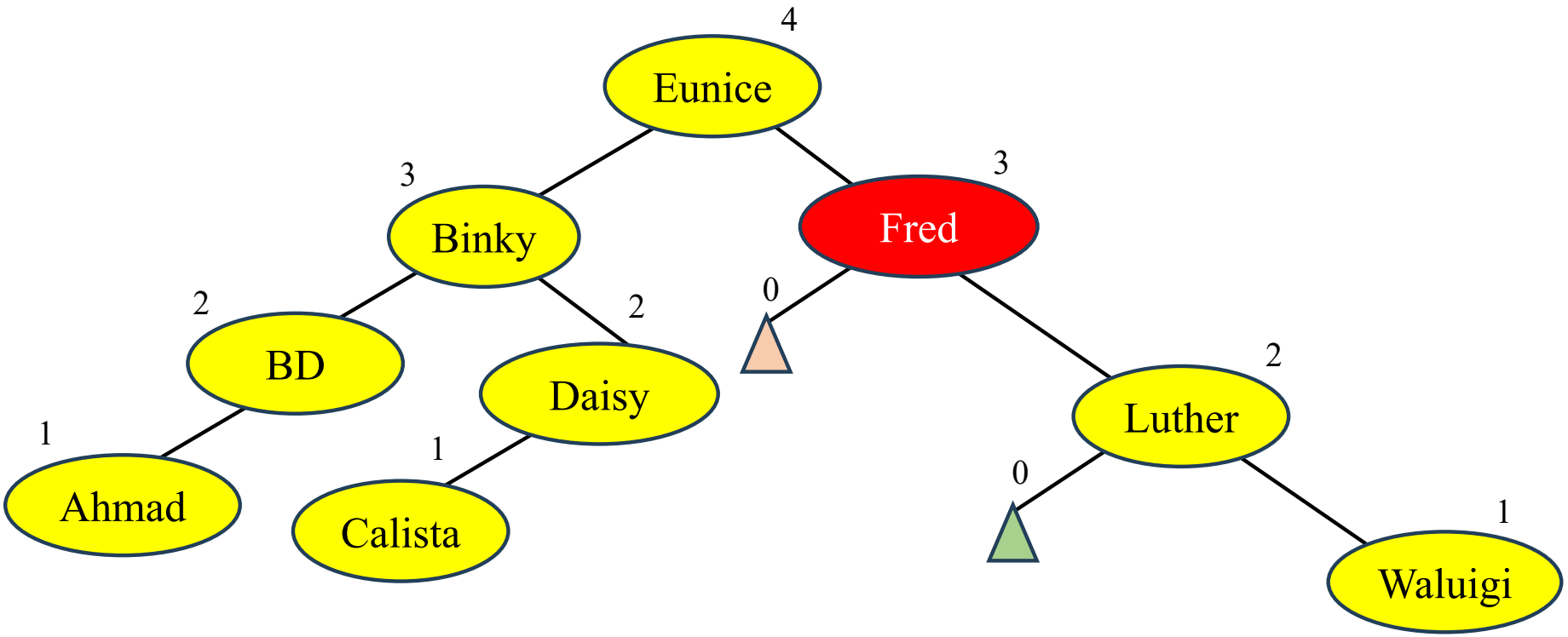
Back to this example.



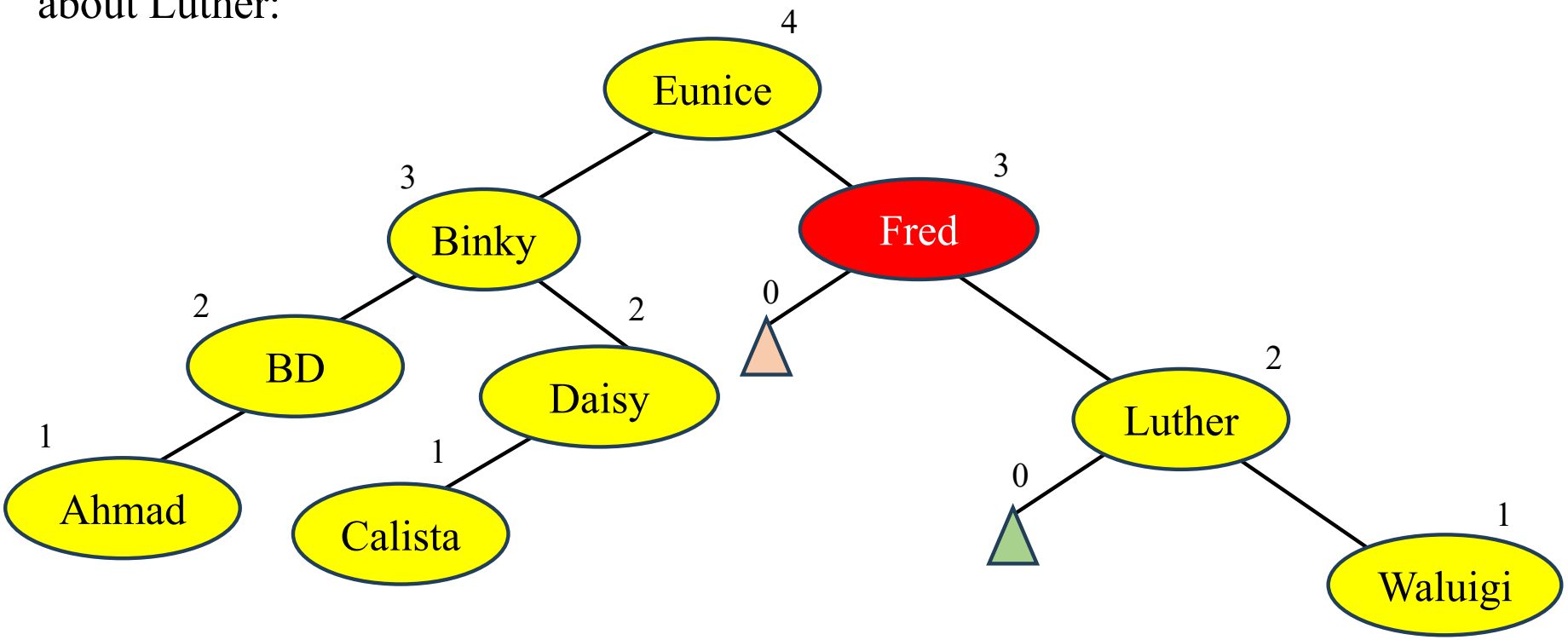
Task: Insert Waluigi.

1. Insert the node.
2. Travel from the new node to the root, adjusting heights and checking for imbalances.

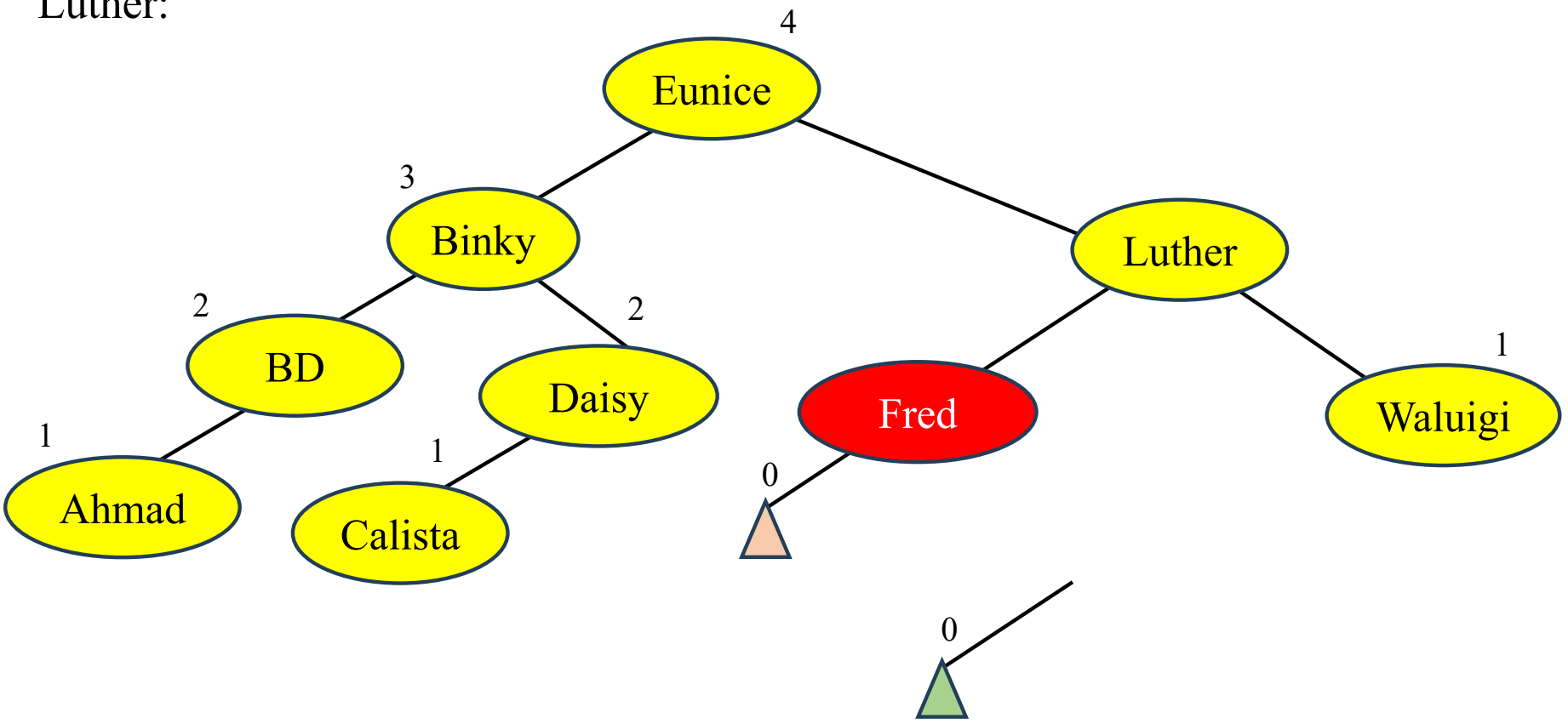
This is a Zig-Zig
(h=3)



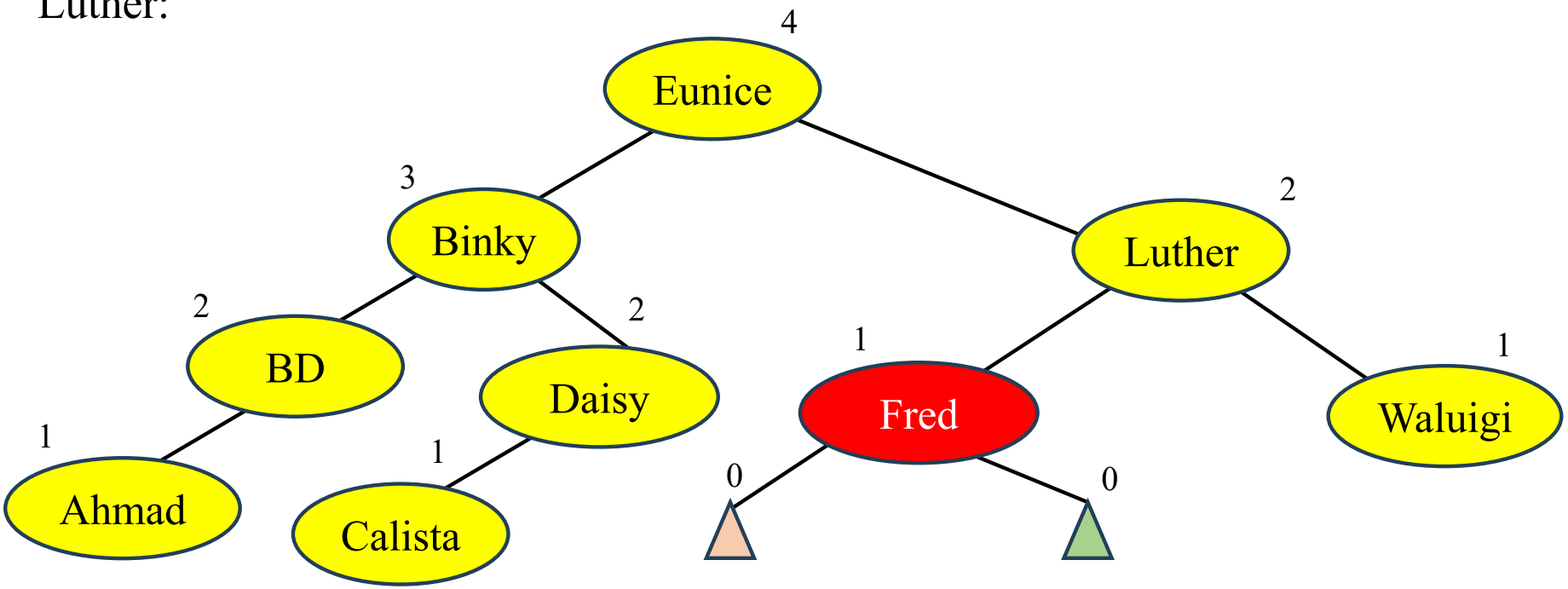
This is a Zig-Zig (h=3)
So fix by rotating
about Luther:



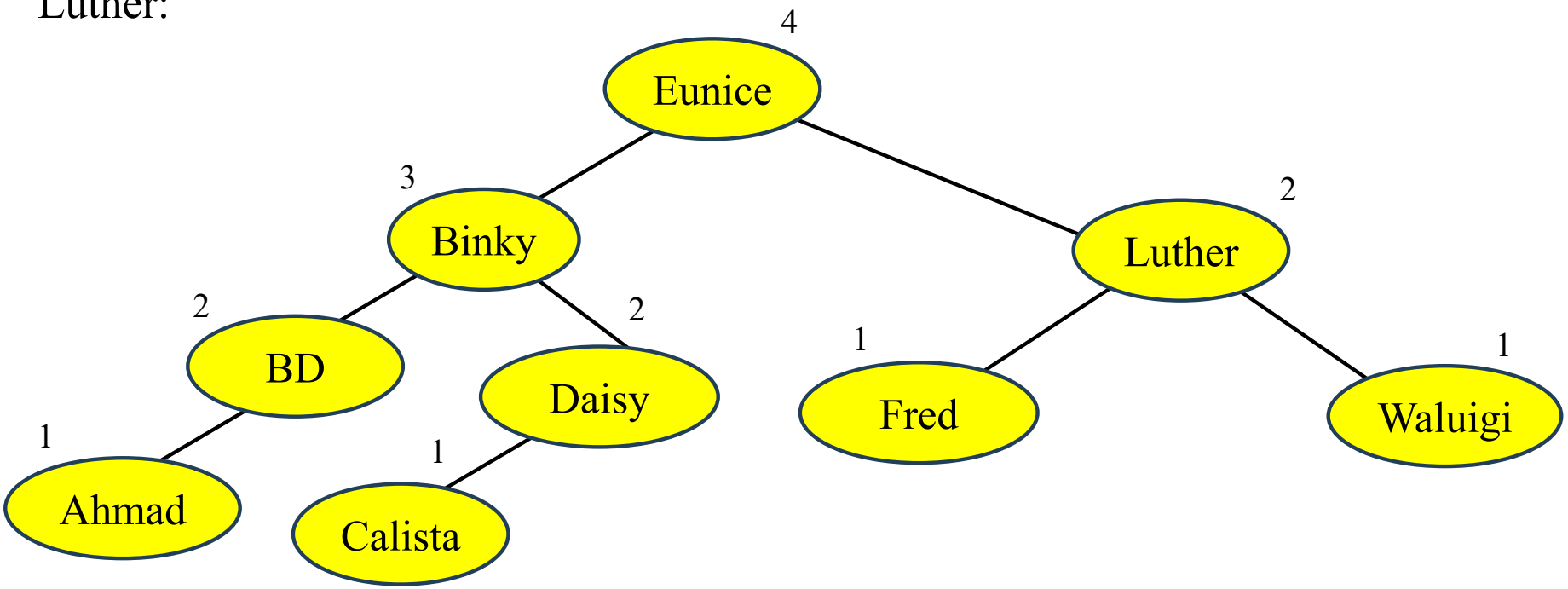
This is a Zig-Zig (h=3)
So fix by rotating about
Luther:



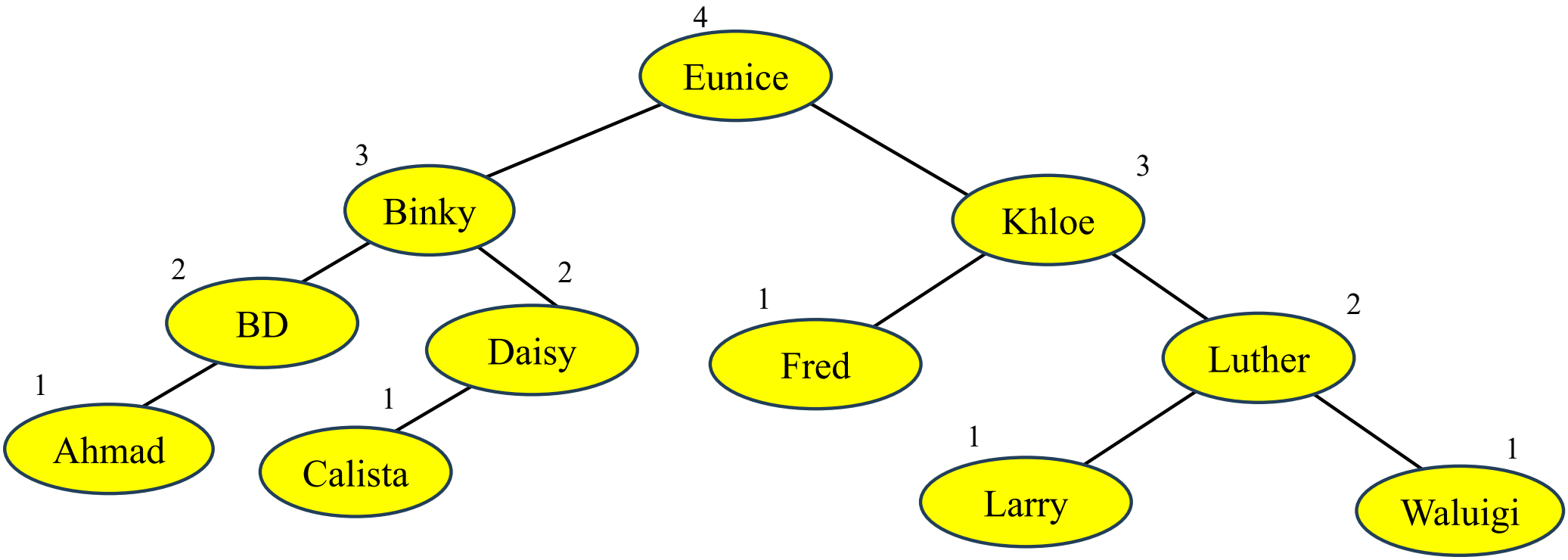
This is a Zig-Zig (h=3)
So fix by rotating about
Luther:



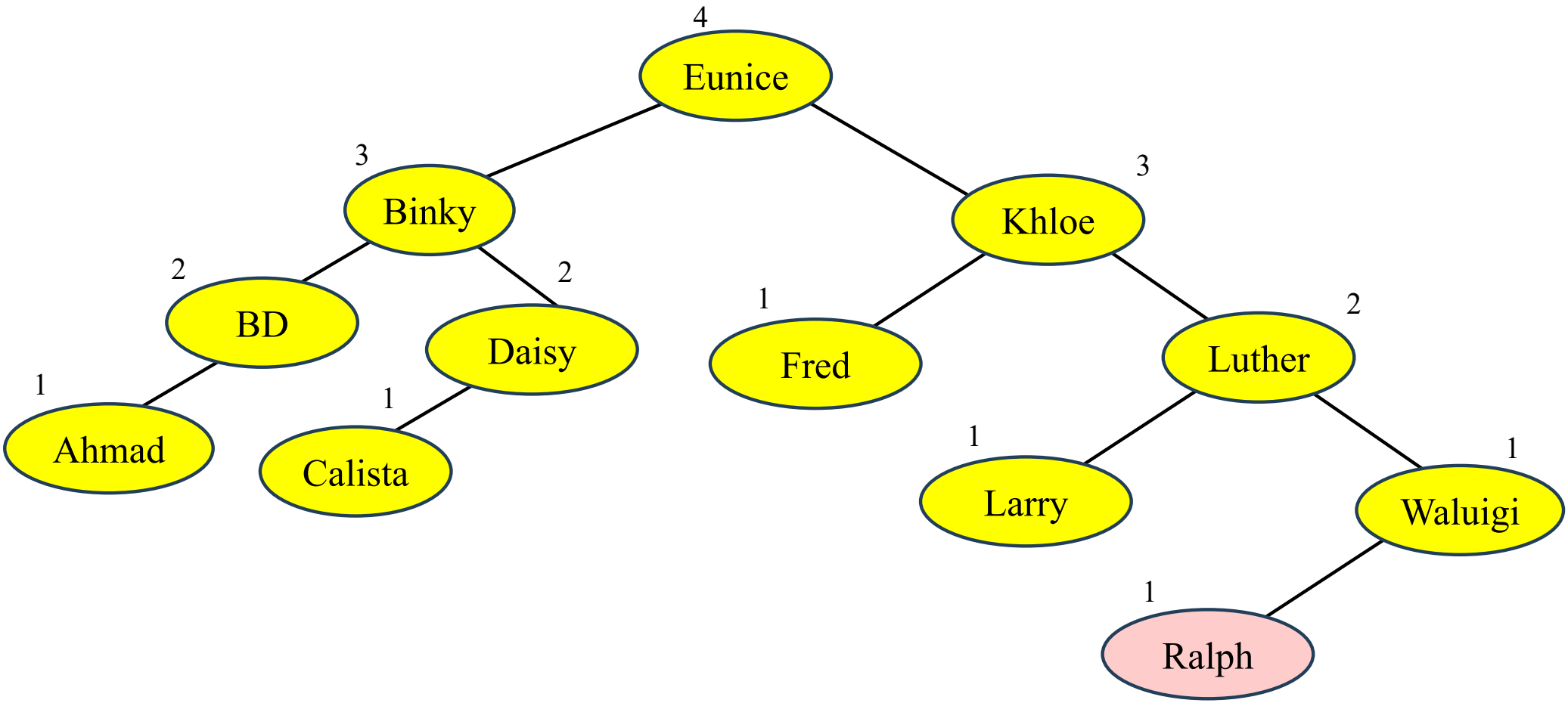
This is a Zig-Zig (h=3)
So fix by rotating about
Luther:



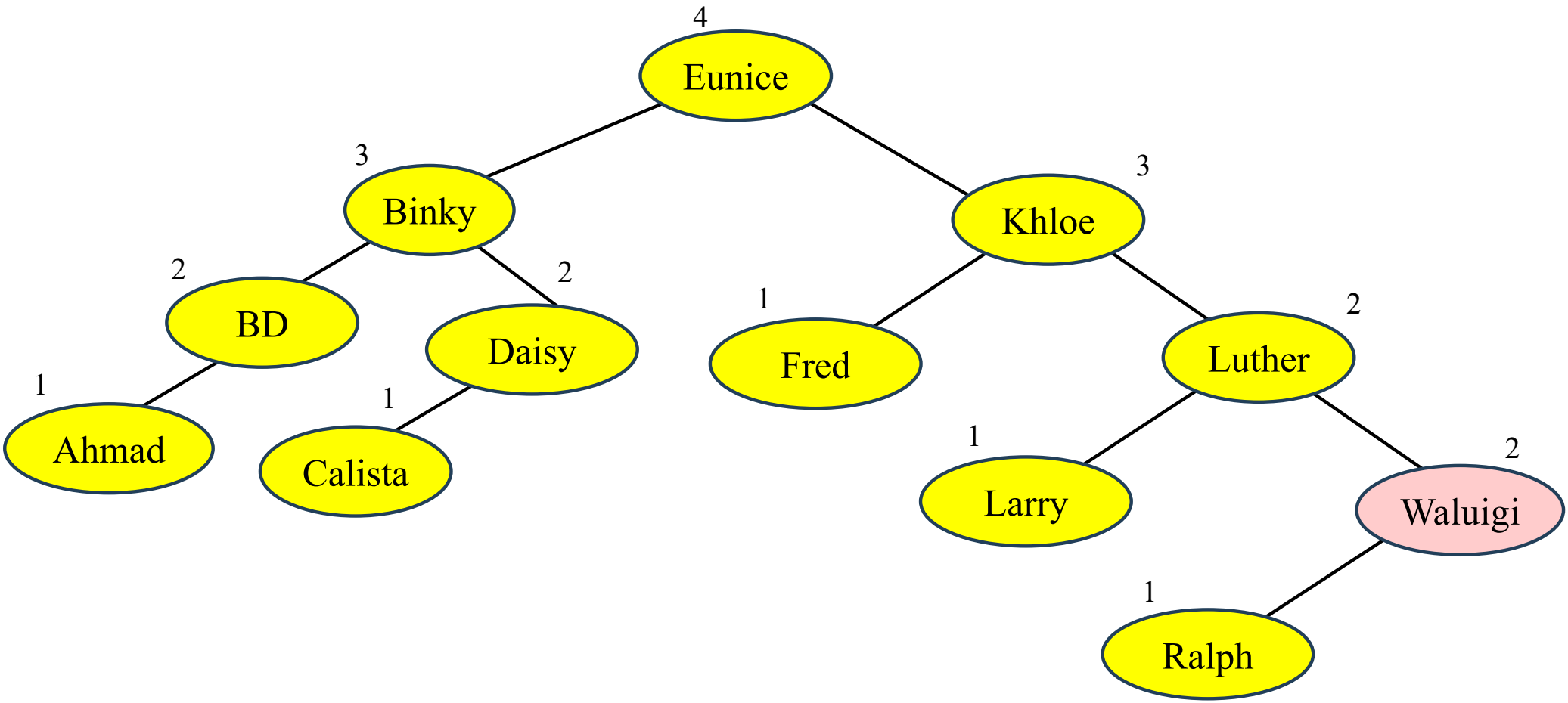
Another example:
Insert Ralph:



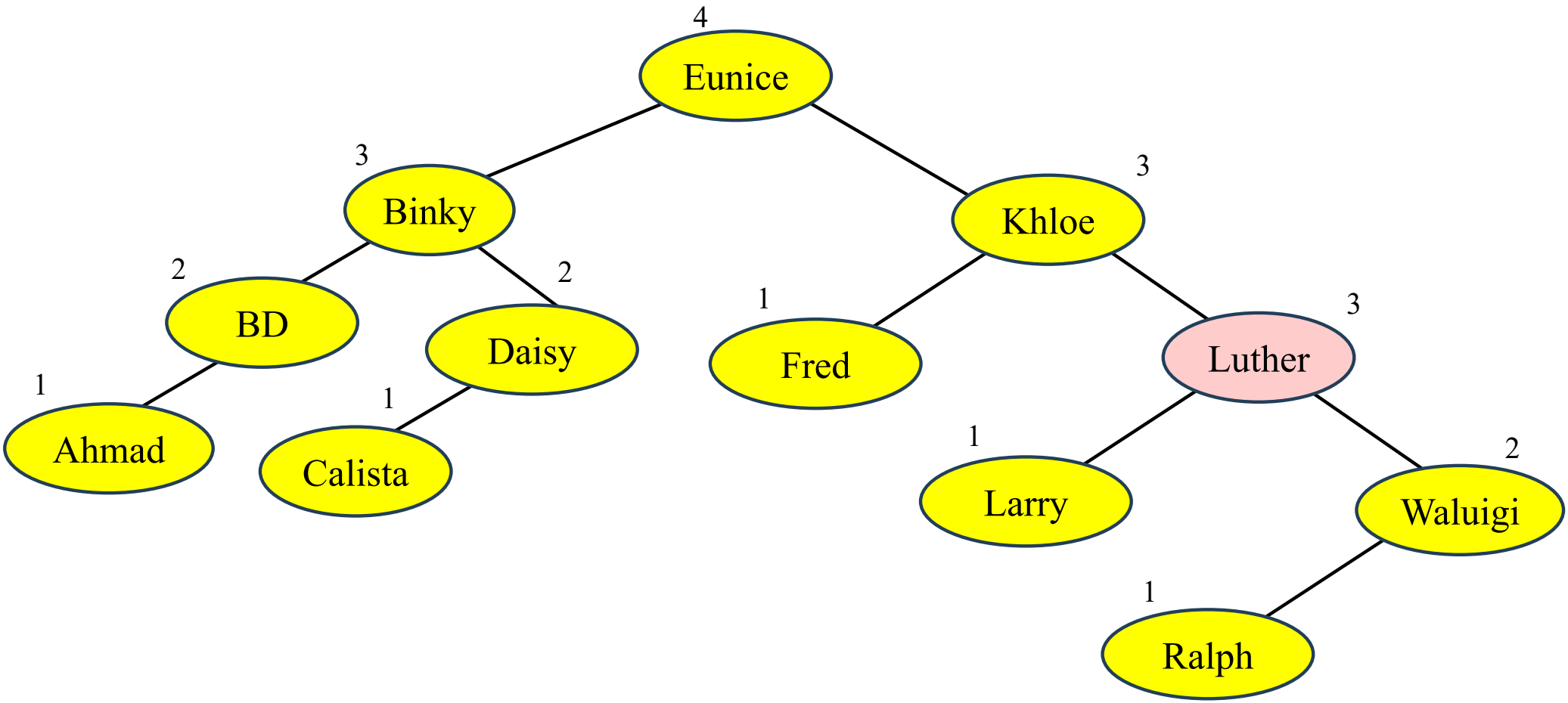
Another example:
Insert Ralph:



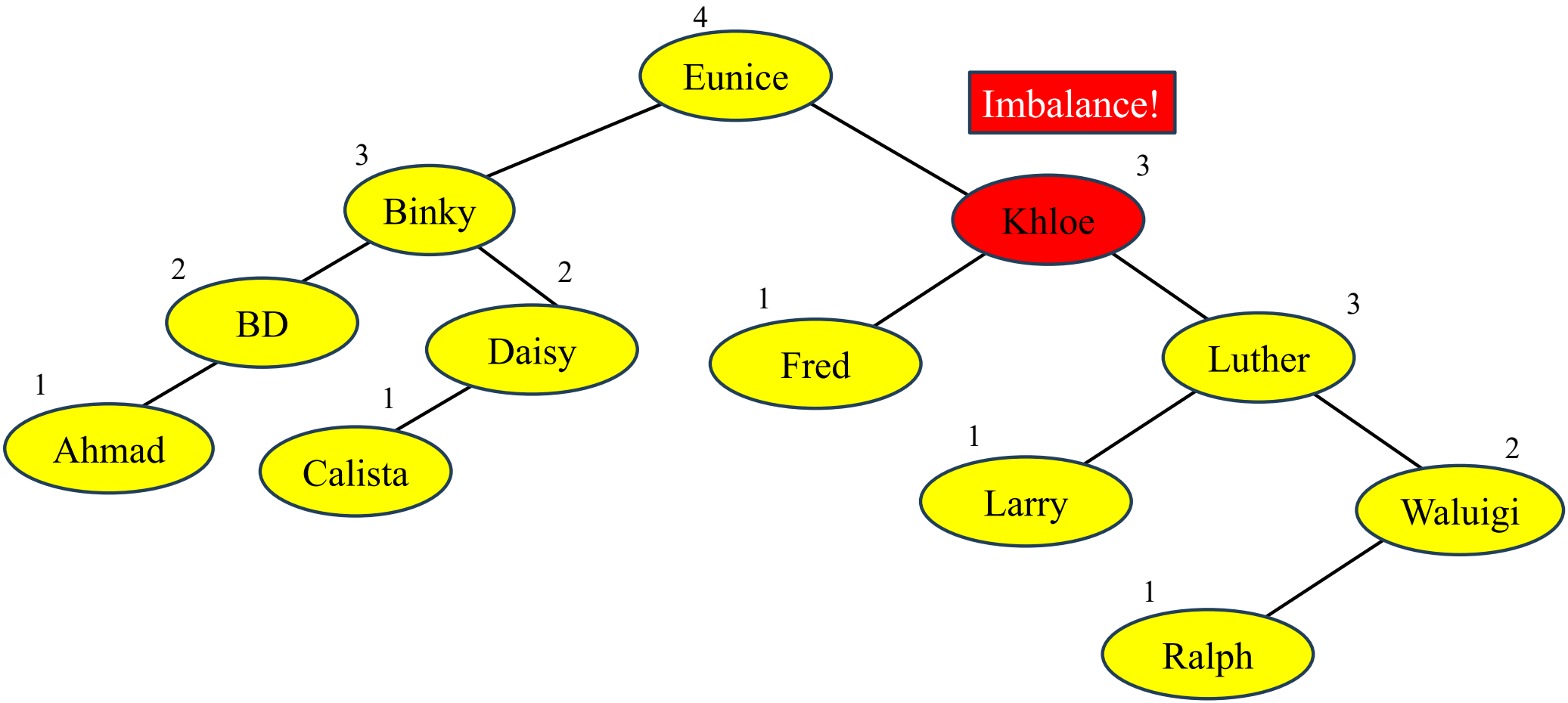
Another example:
Insert Ralph:



Another example:
Insert Ralph:

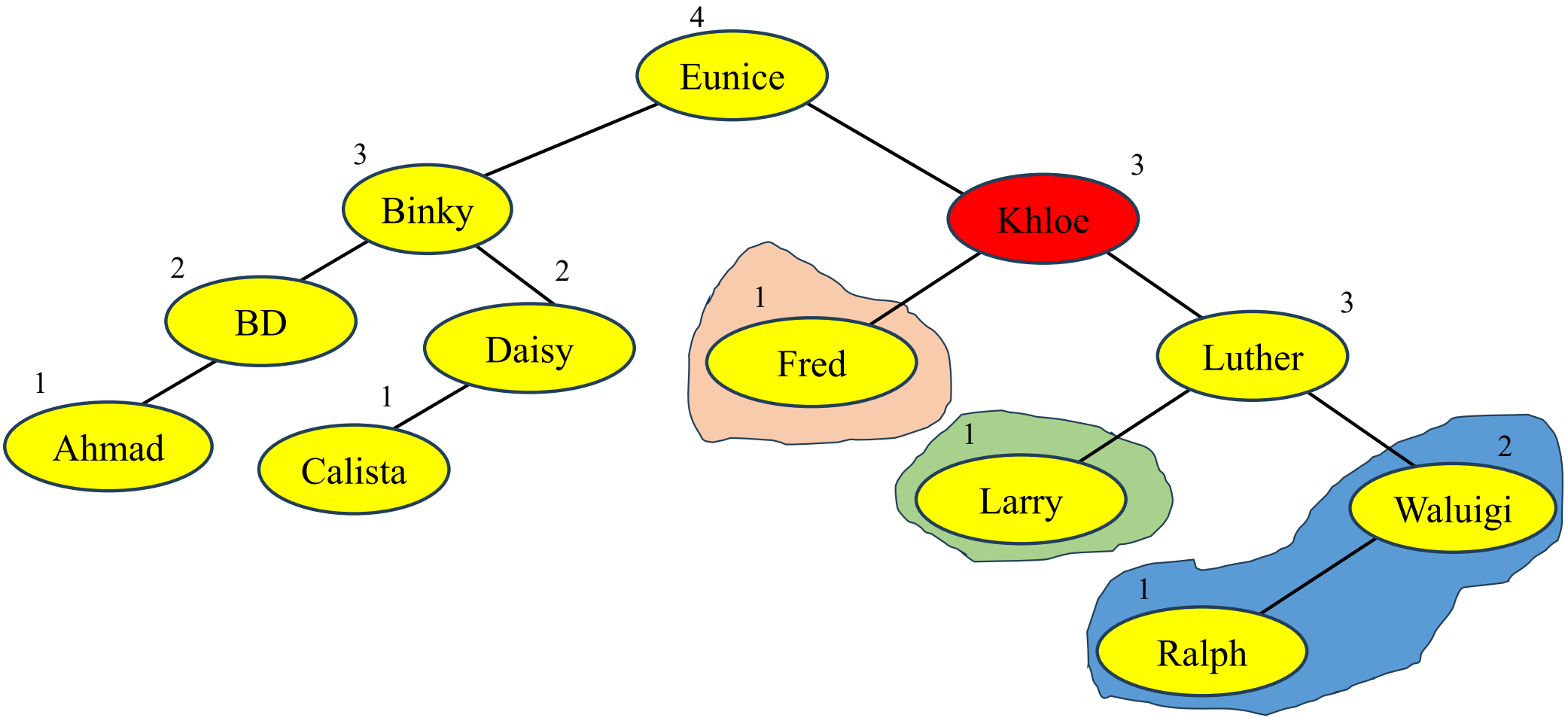


Another example:
Insert Ralph:



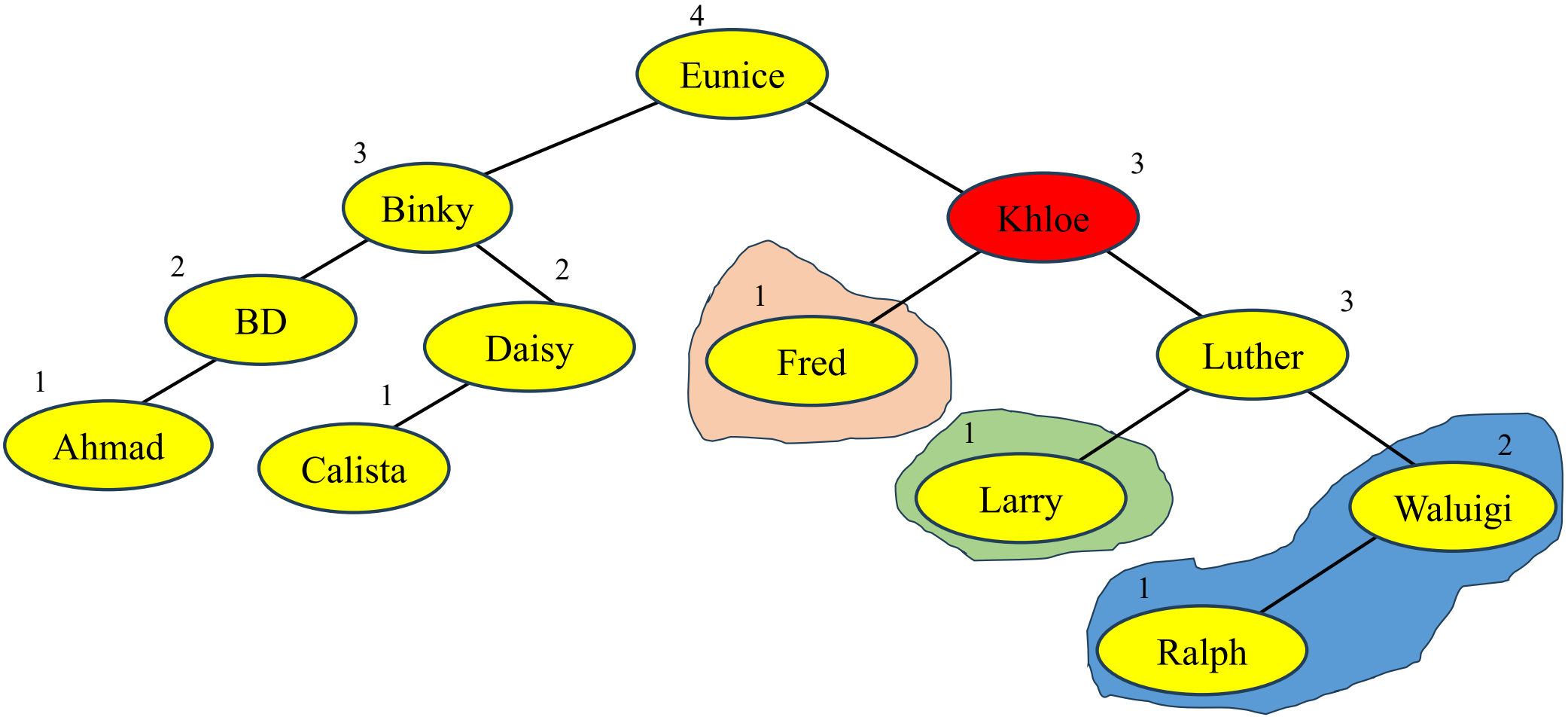
Another example:
Insert Ralph:

It's a Zig-Zig.



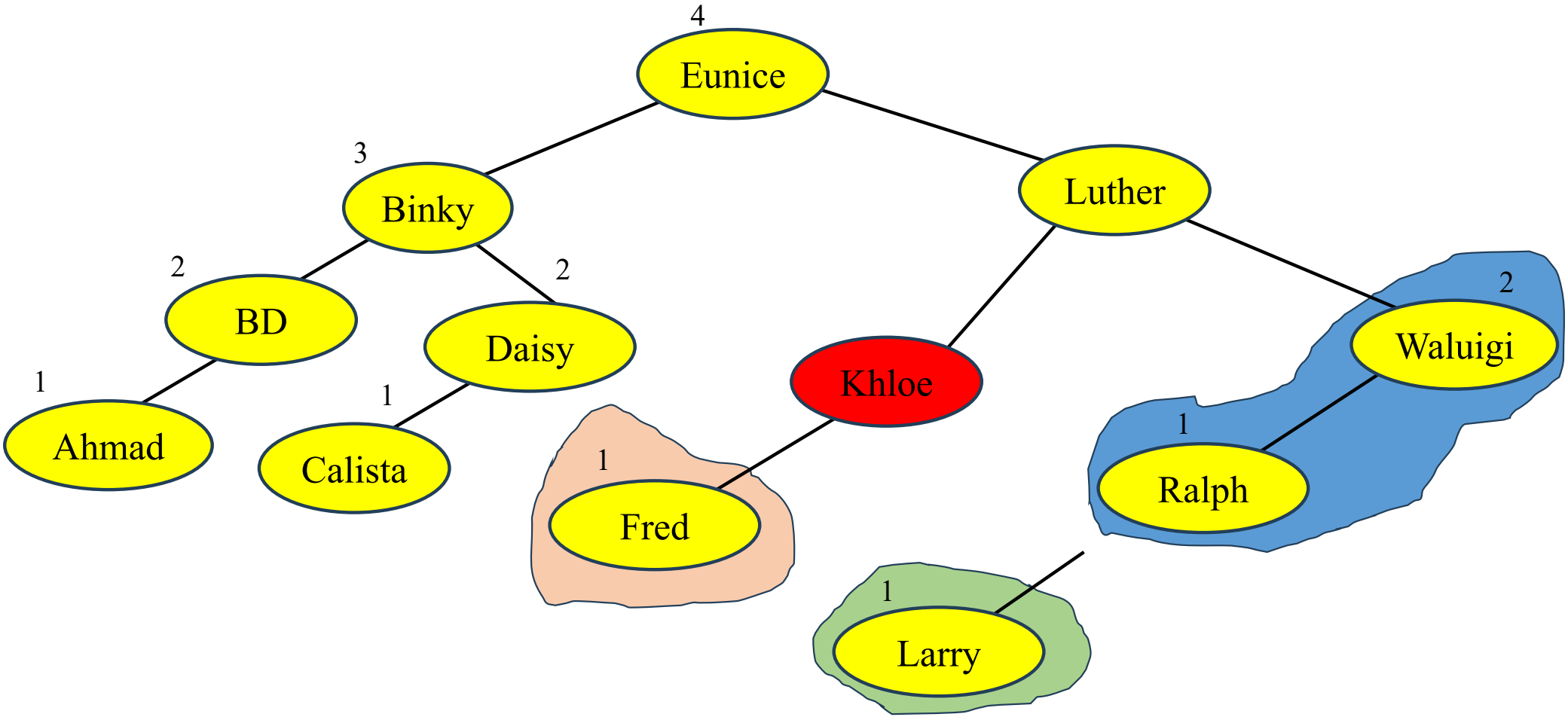
Another example:
Insert Ralph:

It's a Zig-Zig
So rotate about Luther.



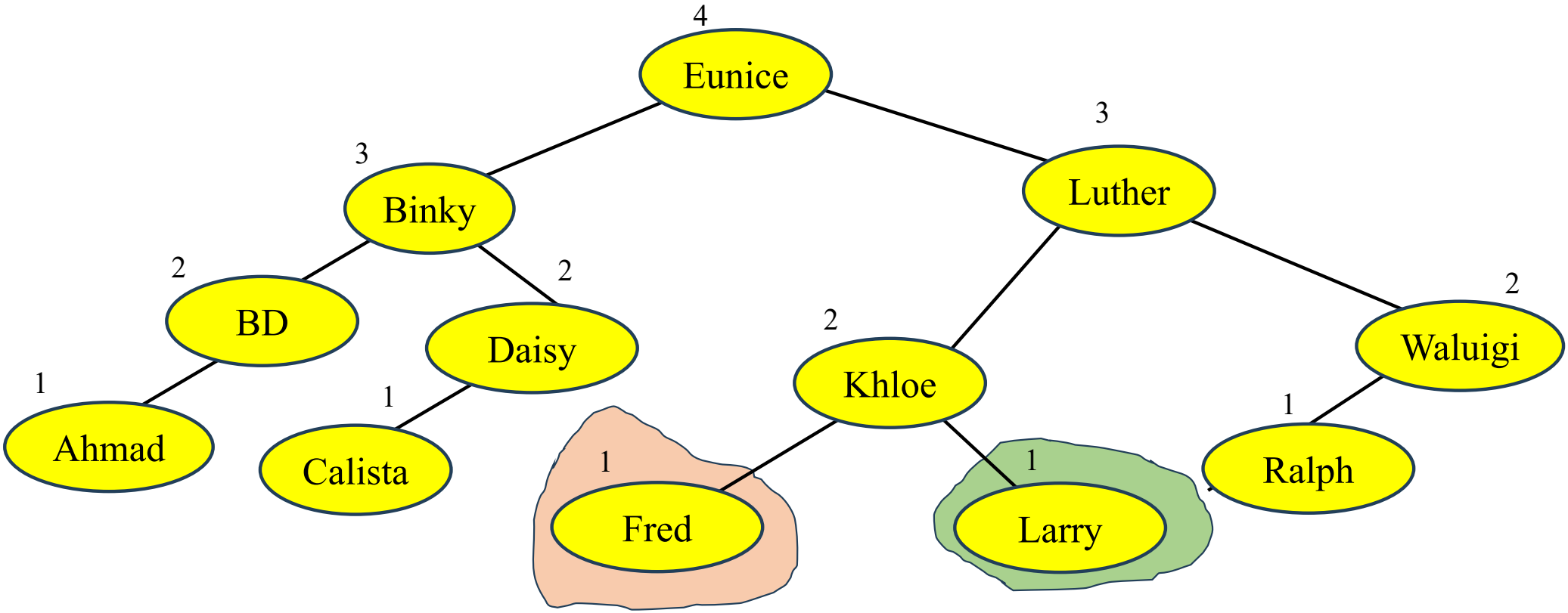
Another example:
Insert Ralph:

It's a Zig-Zig
So rotate about Luther.

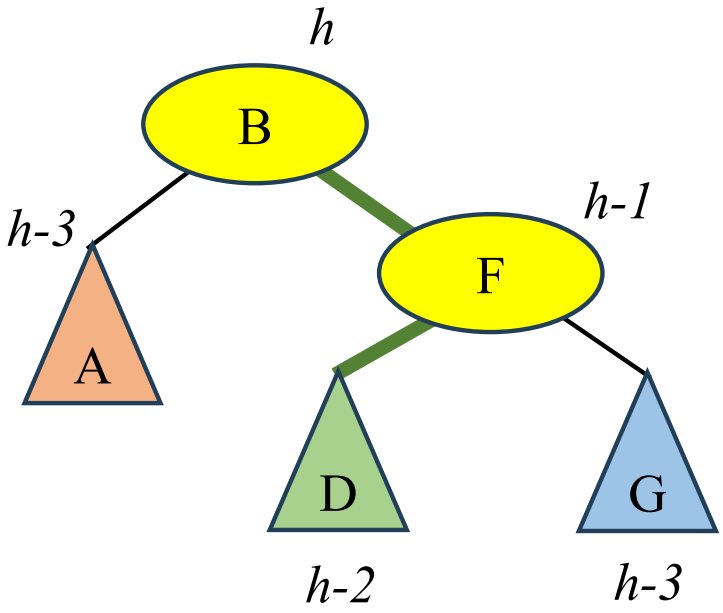


Another example:
Insert Ralph:

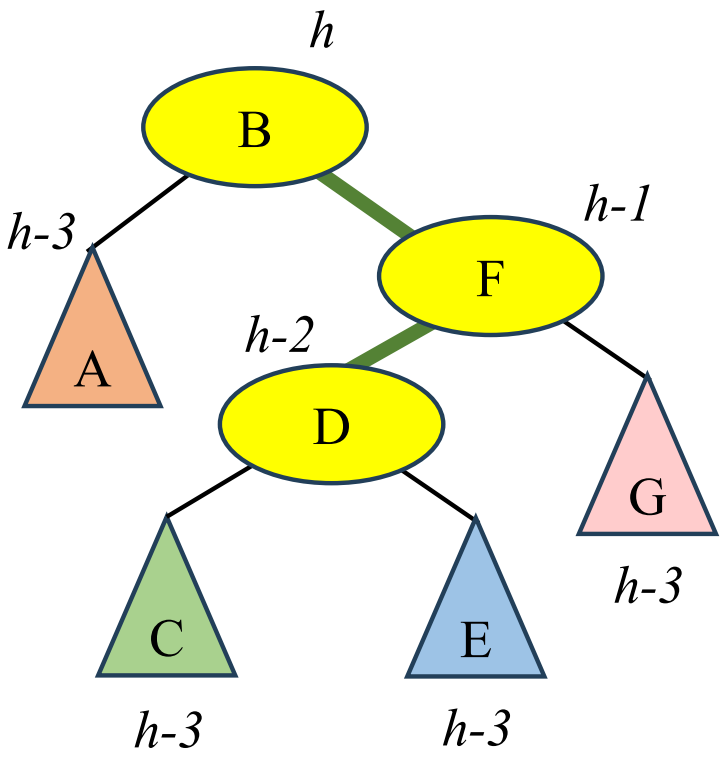
It's a Zig-Zig
So rotate about Luther.

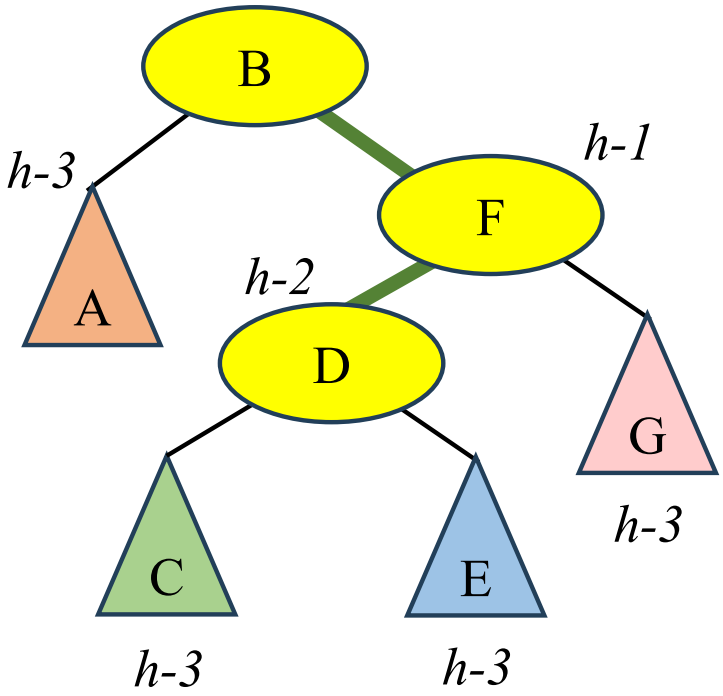


Fixing Zig-Zag



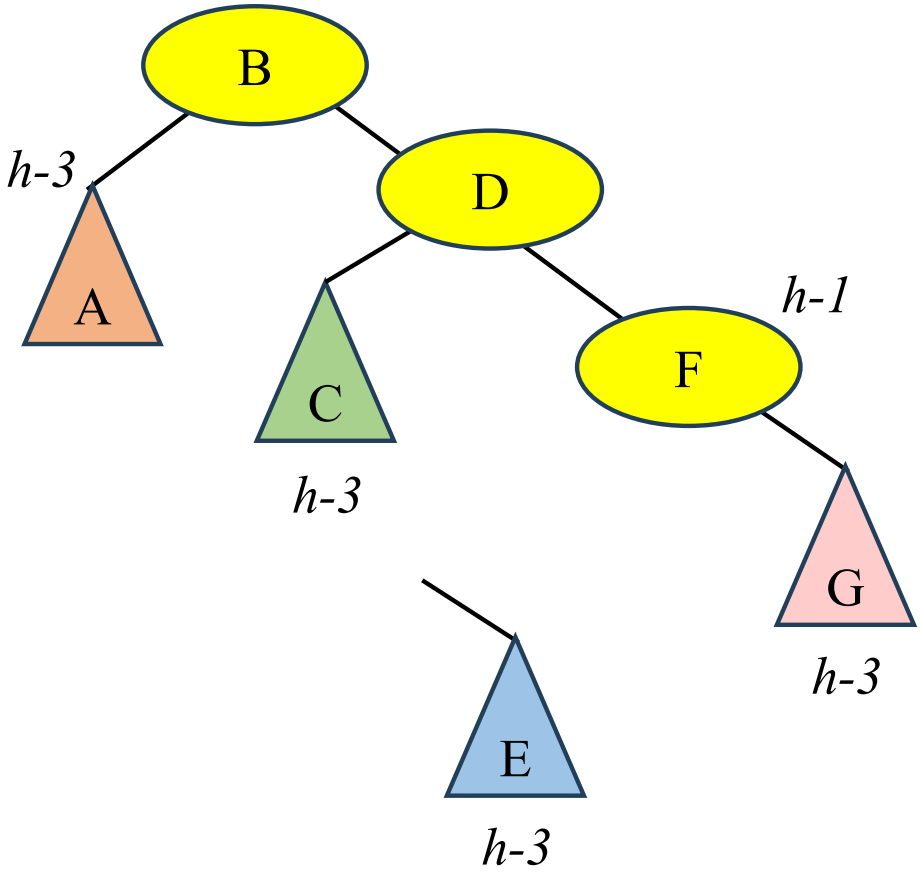
Expand the D subtree and relabel:

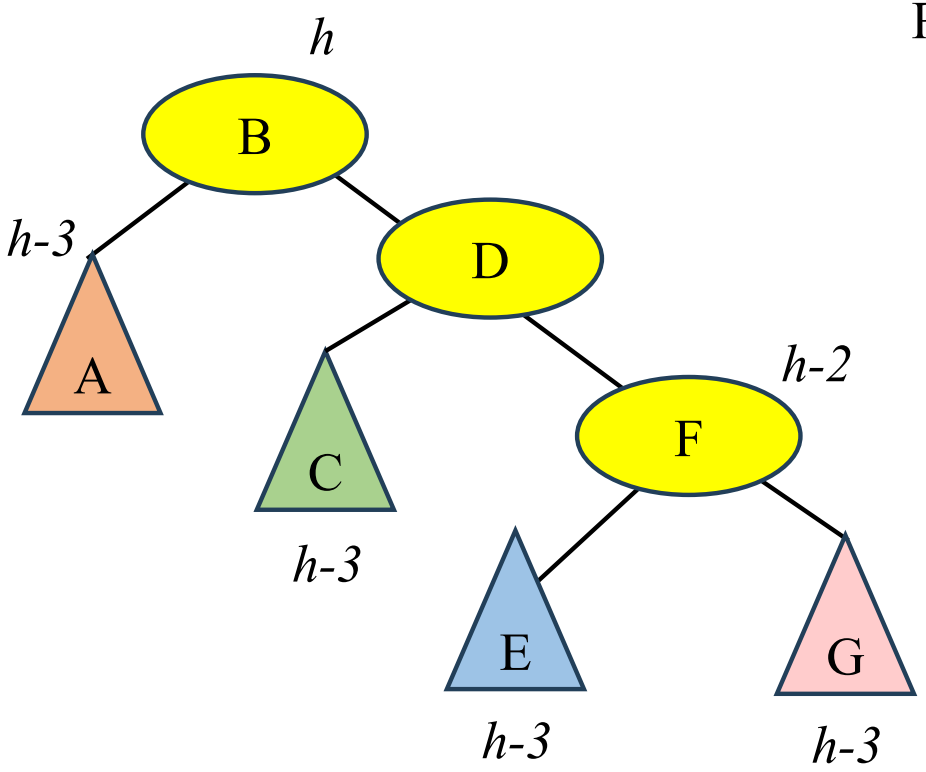

h
B
h-3
h-1
F
A
h-2
D
G
h-3
C
E
h-3
h-3



<u>To fix Zig-Zag:</u>
Rotate twice about the grandchild (D).

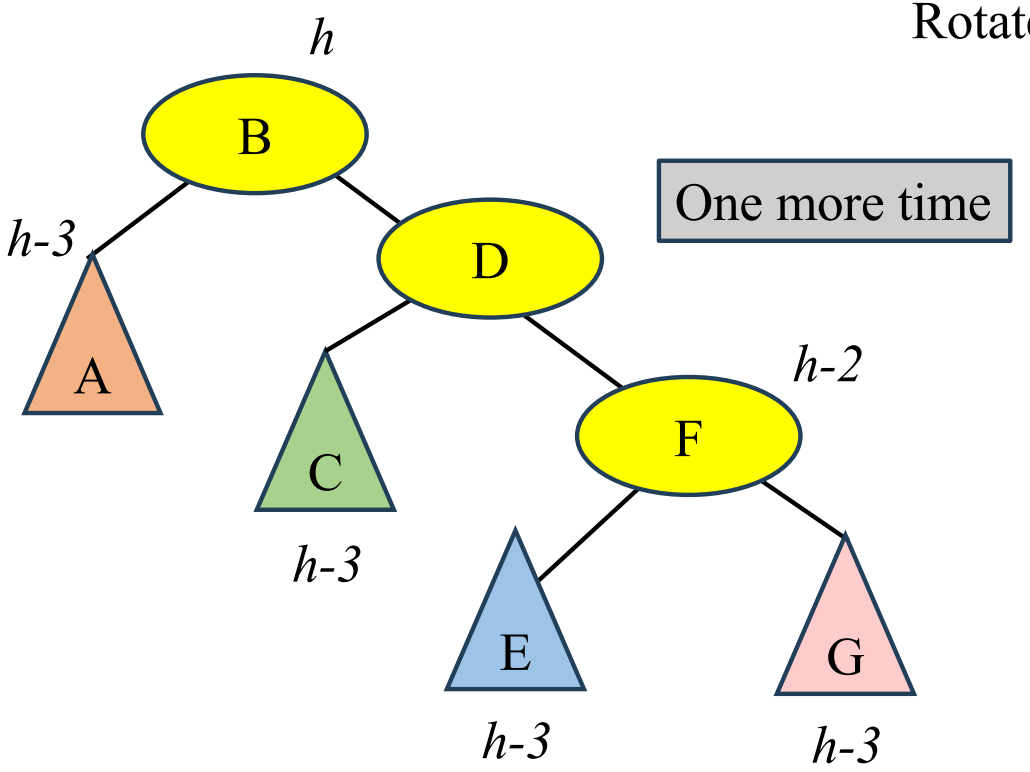
<u>To fix Zig-Zag:</u>
Rotate twice about the grandchild (D).

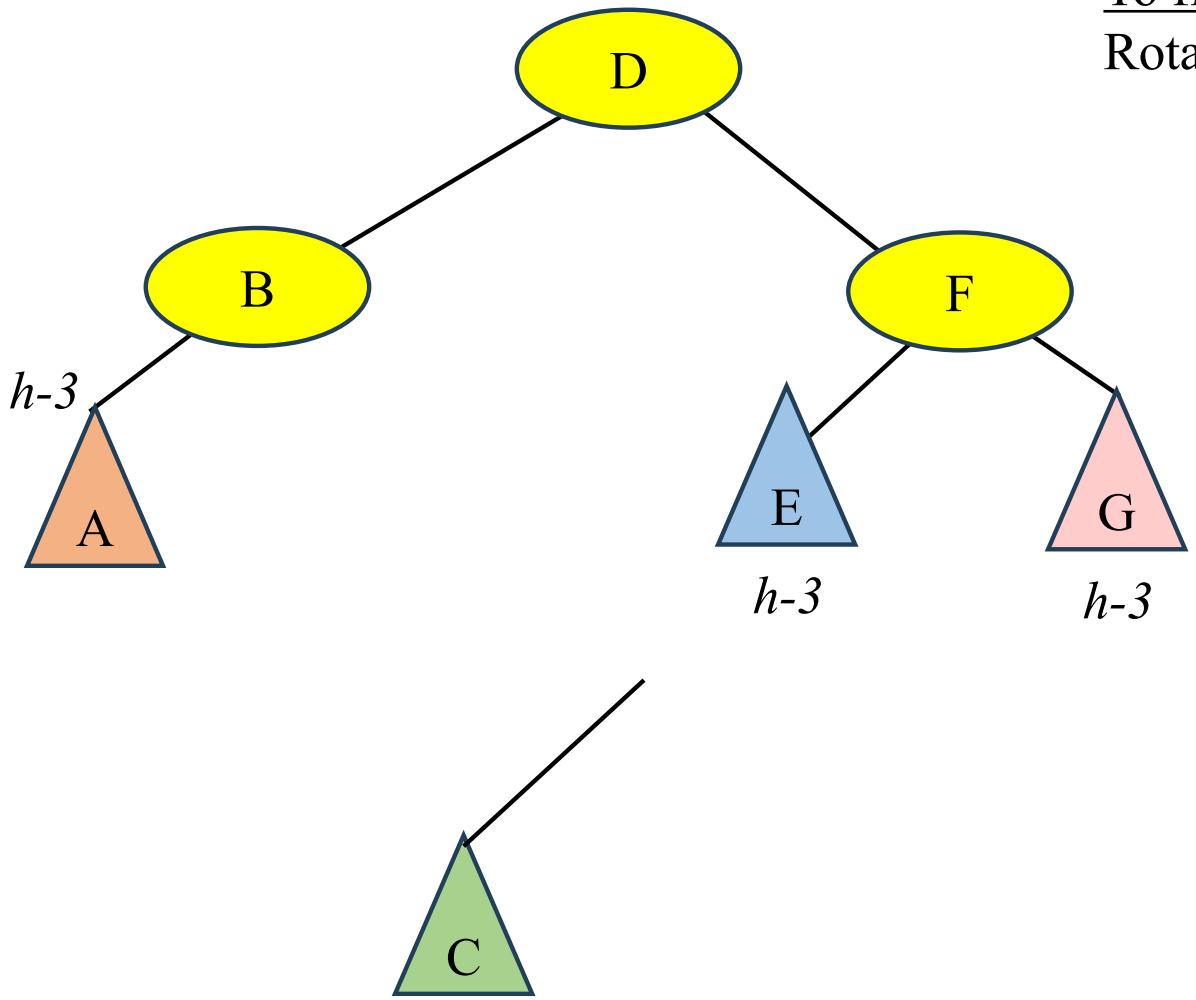




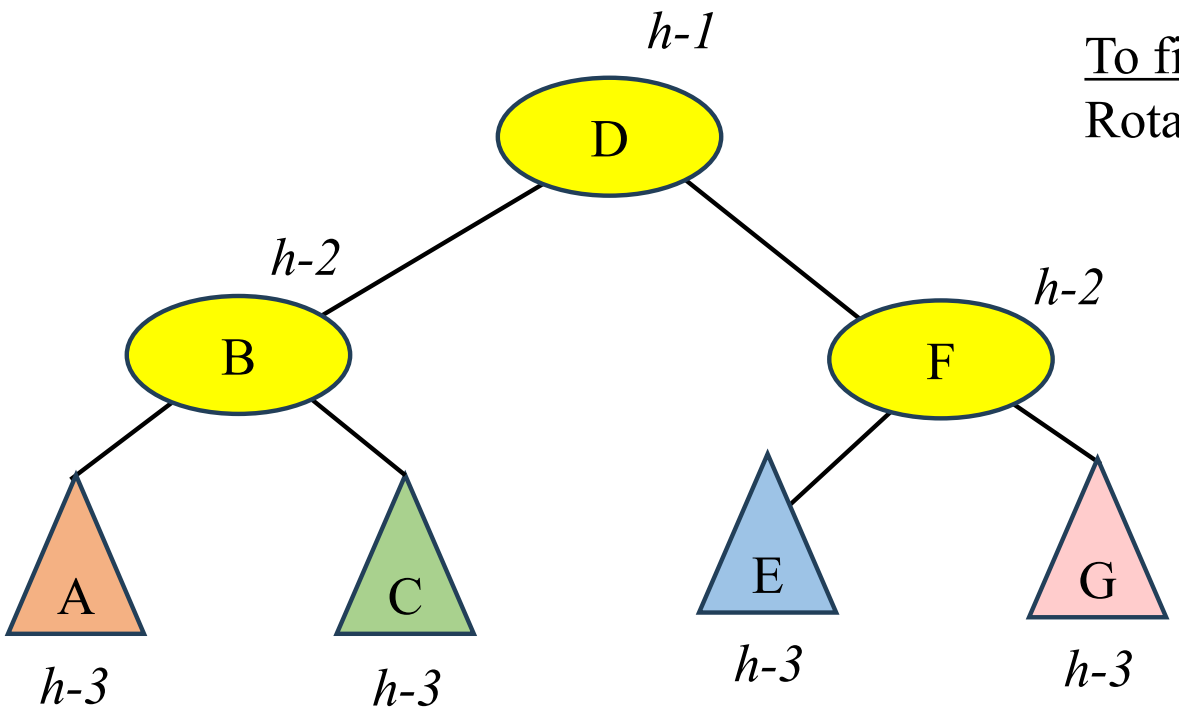
<u>To fix Zig-Zag:</u>
Rotate twice about the grandchild (D).

<u>To fix Zig-Zag:</u>
Rotate twice about the grandchild (D).





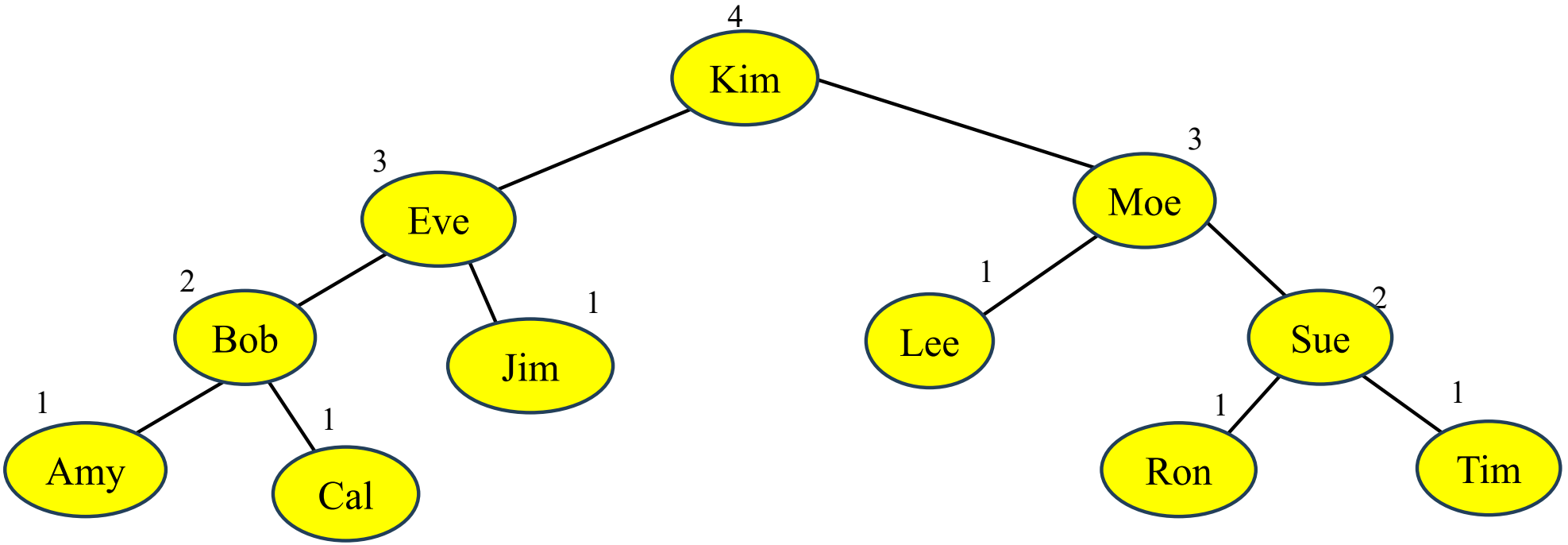
<u>To fix Zig-Zag:</u>
Rotate twice about the grandchild (D).



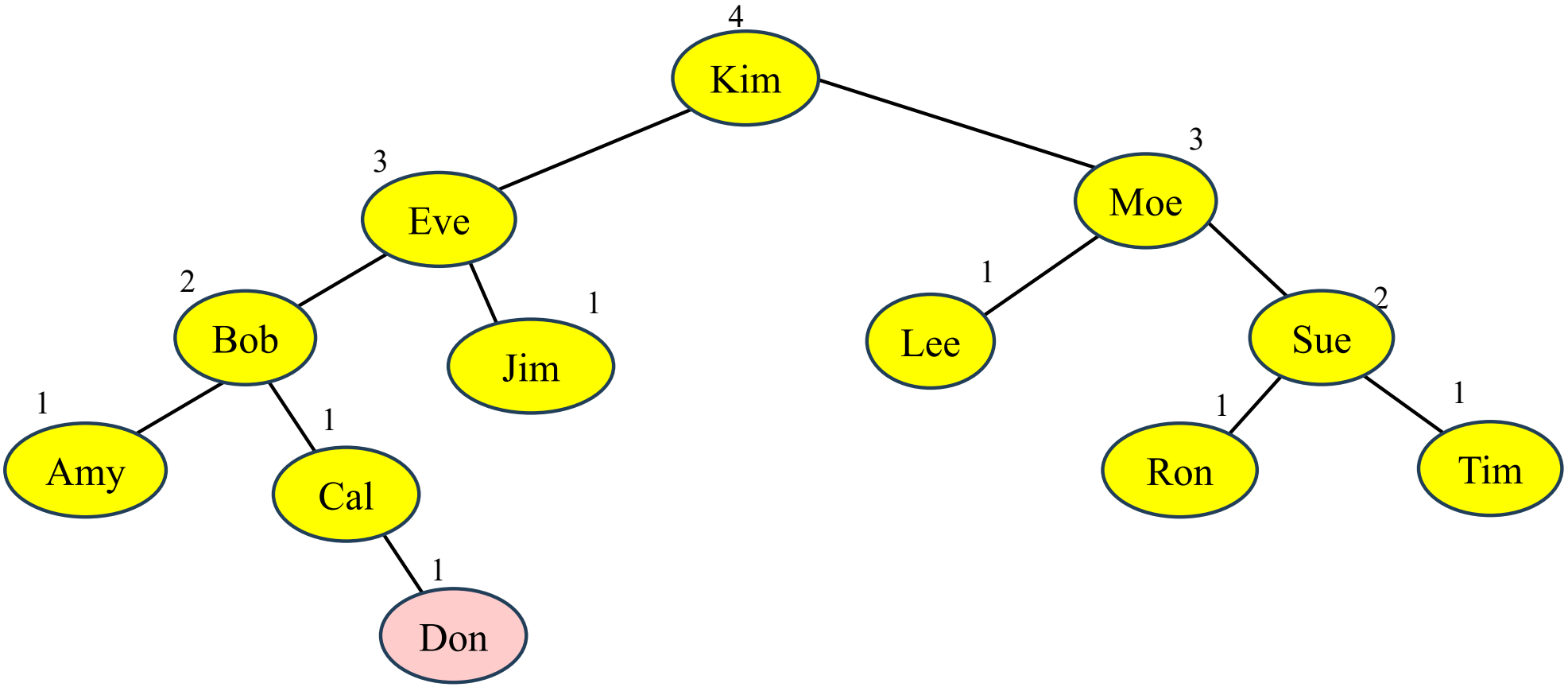
<u>To fix Zig-Zag:</u>
Rotate twice about the grandchild (D).

Example:
Insert Don:

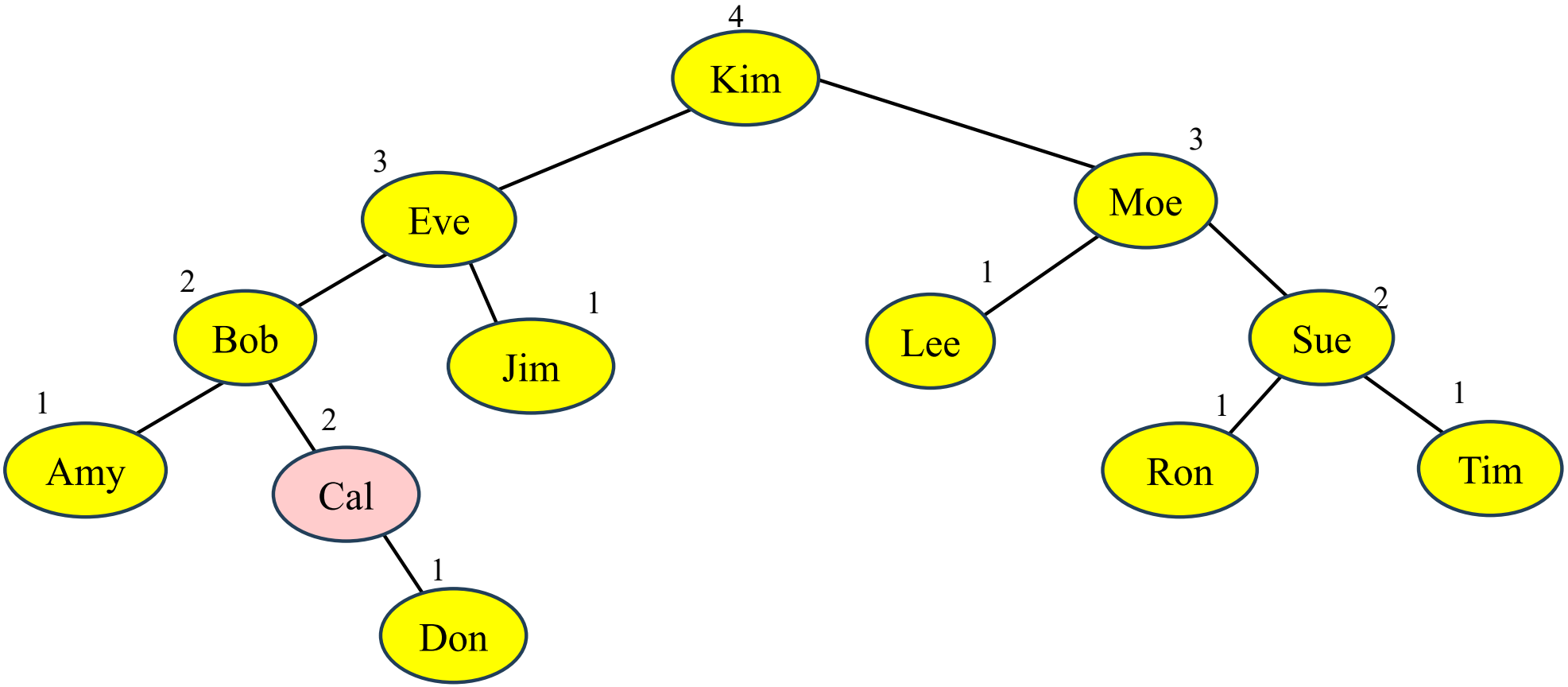
It's a Zig-Zig
So rotate about Luther.



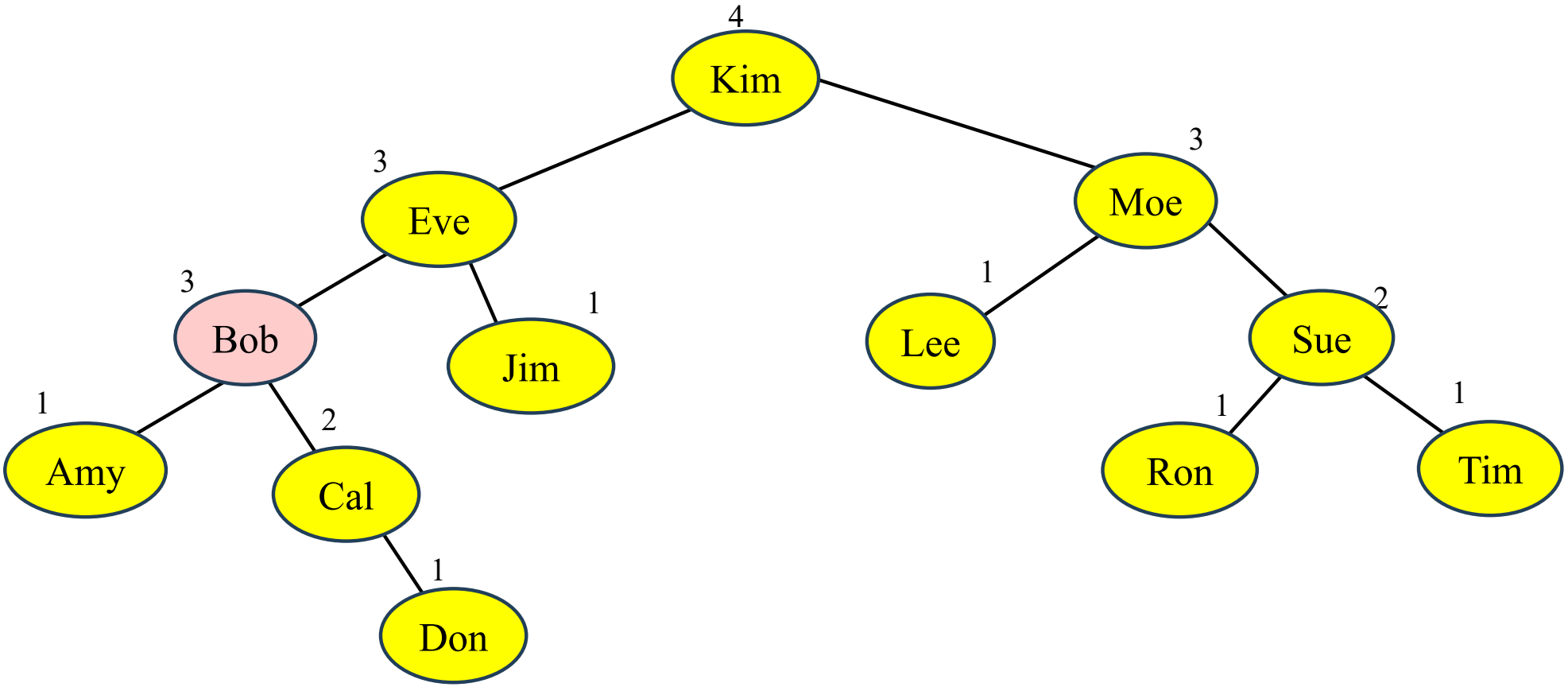
Example:
Insert Don:



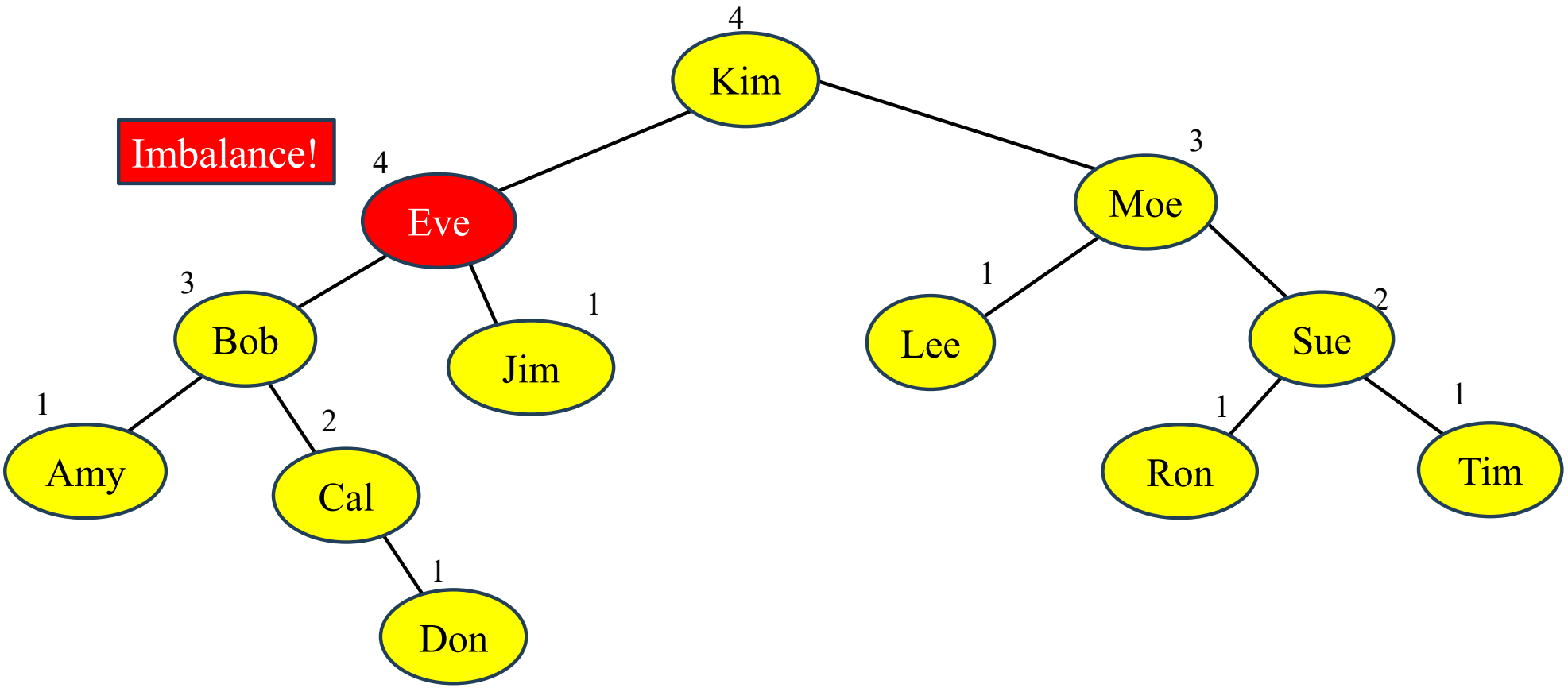
Example:
Insert Don:



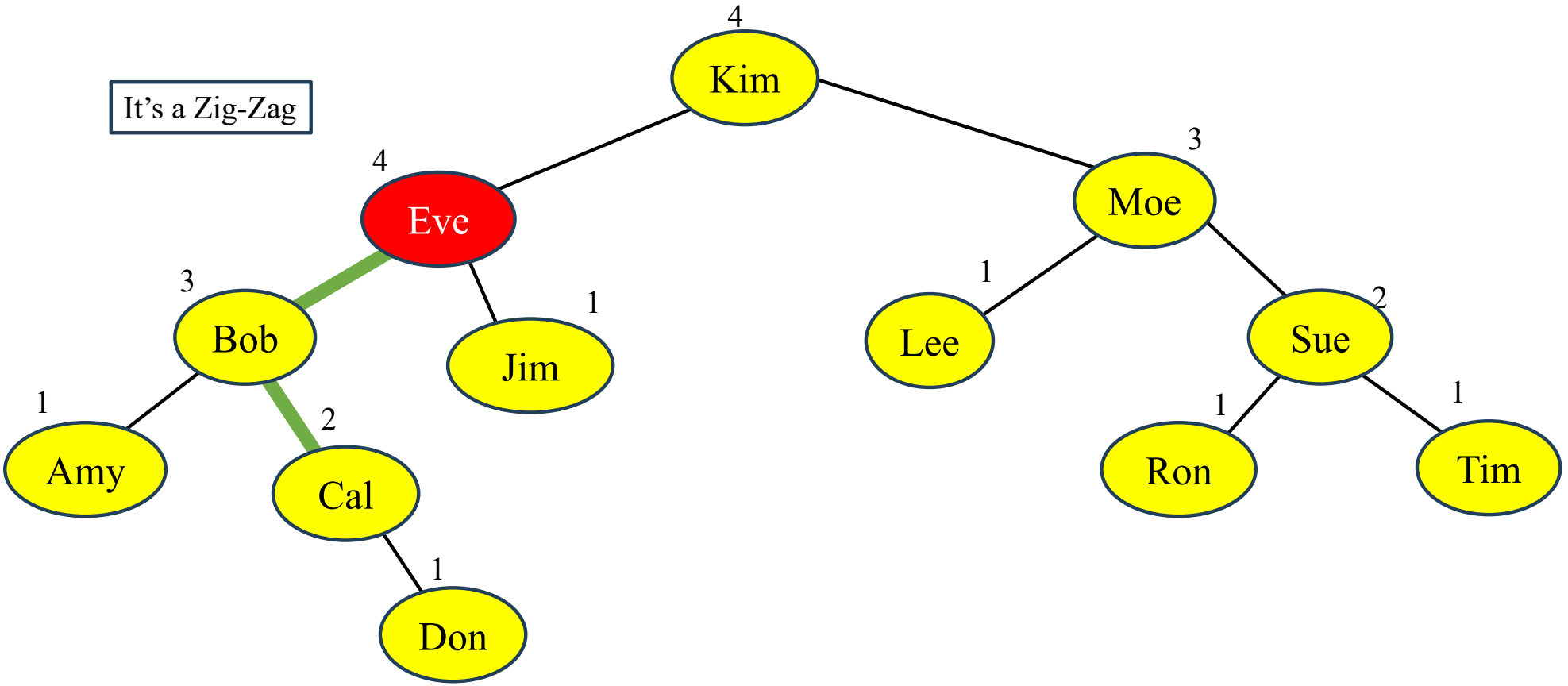
Example:
Insert Don:



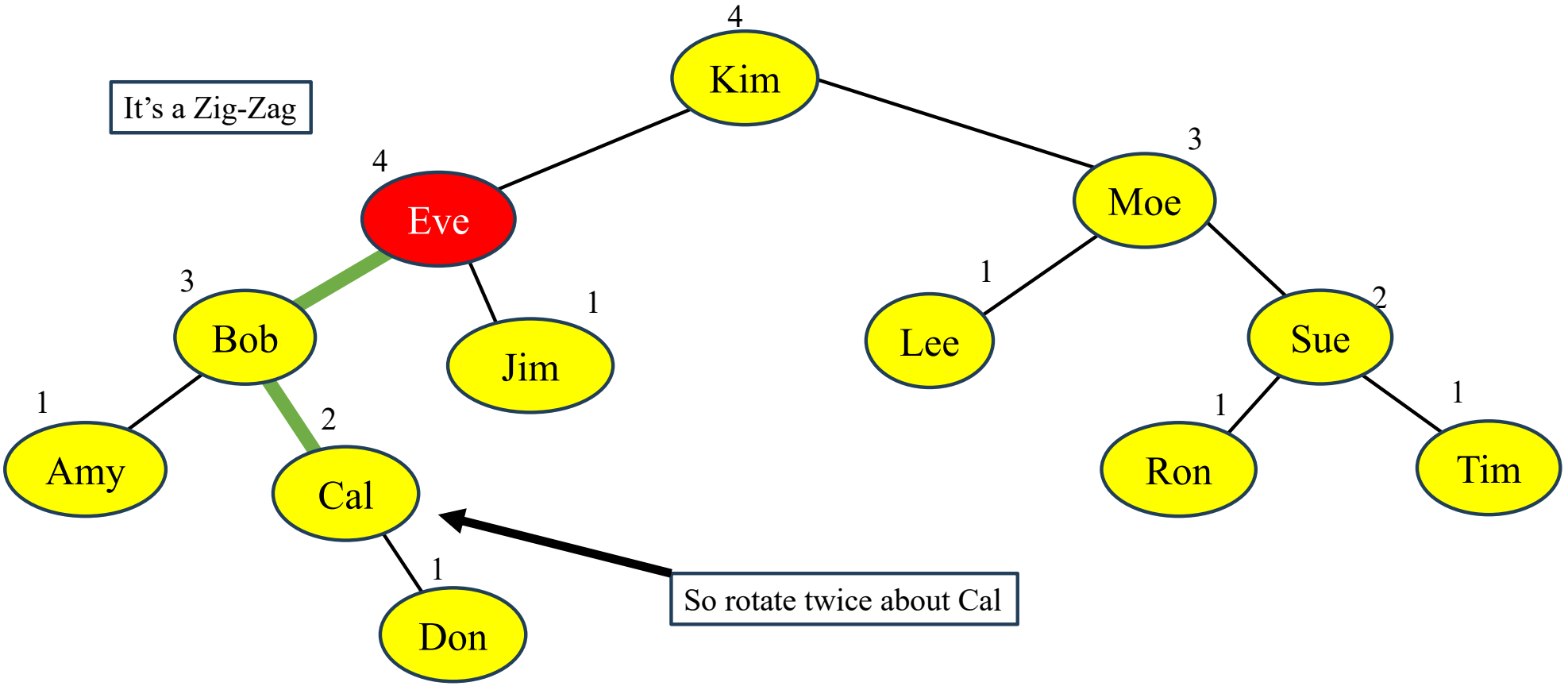
Example:
Insert Don:



Example:
Insert Don:

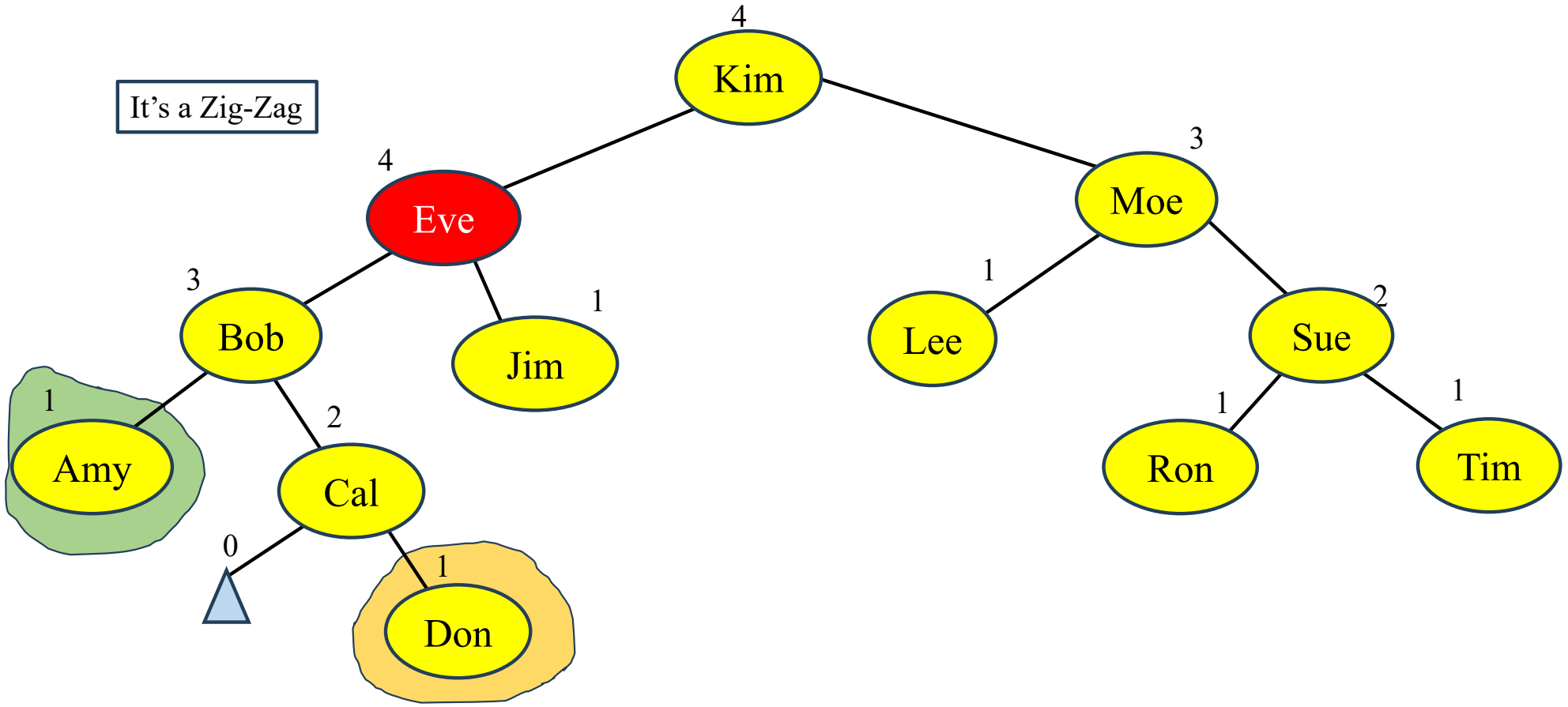


Example:
Insert Don:



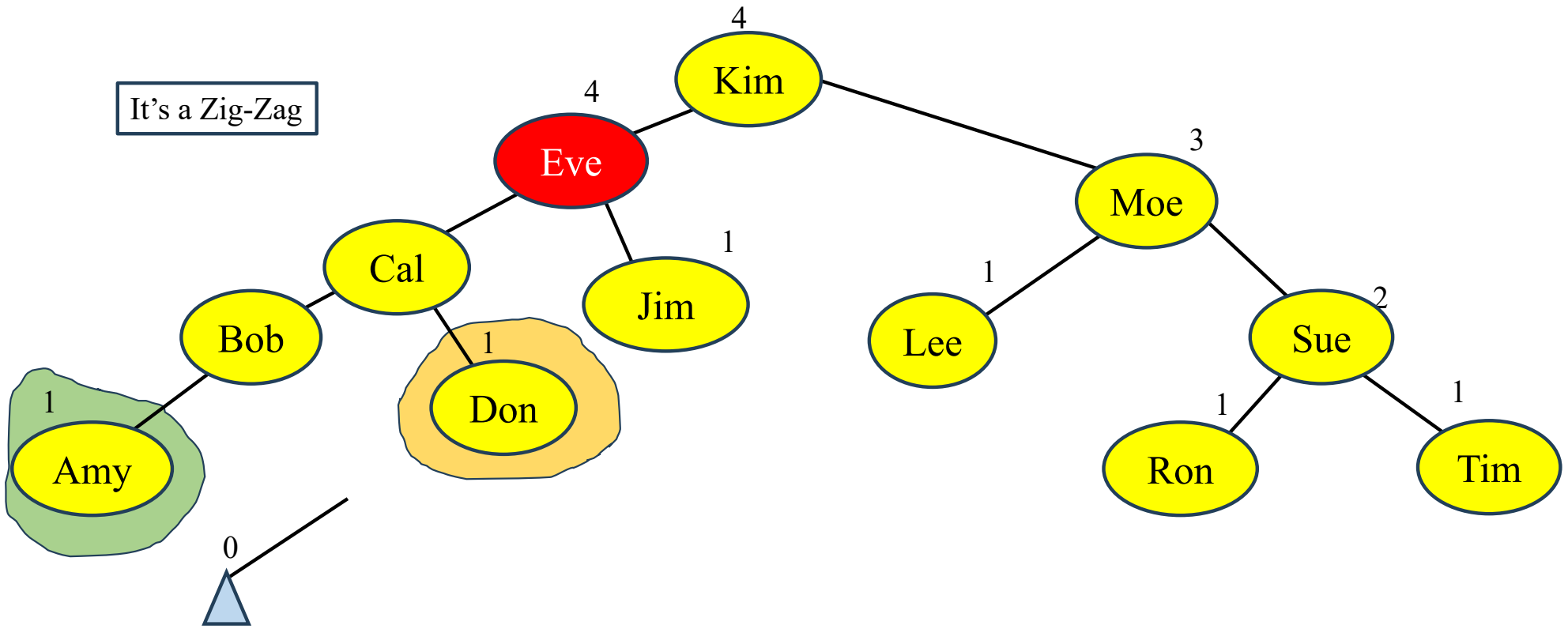
Example:
Insert Don:

Rotate twice about Cal



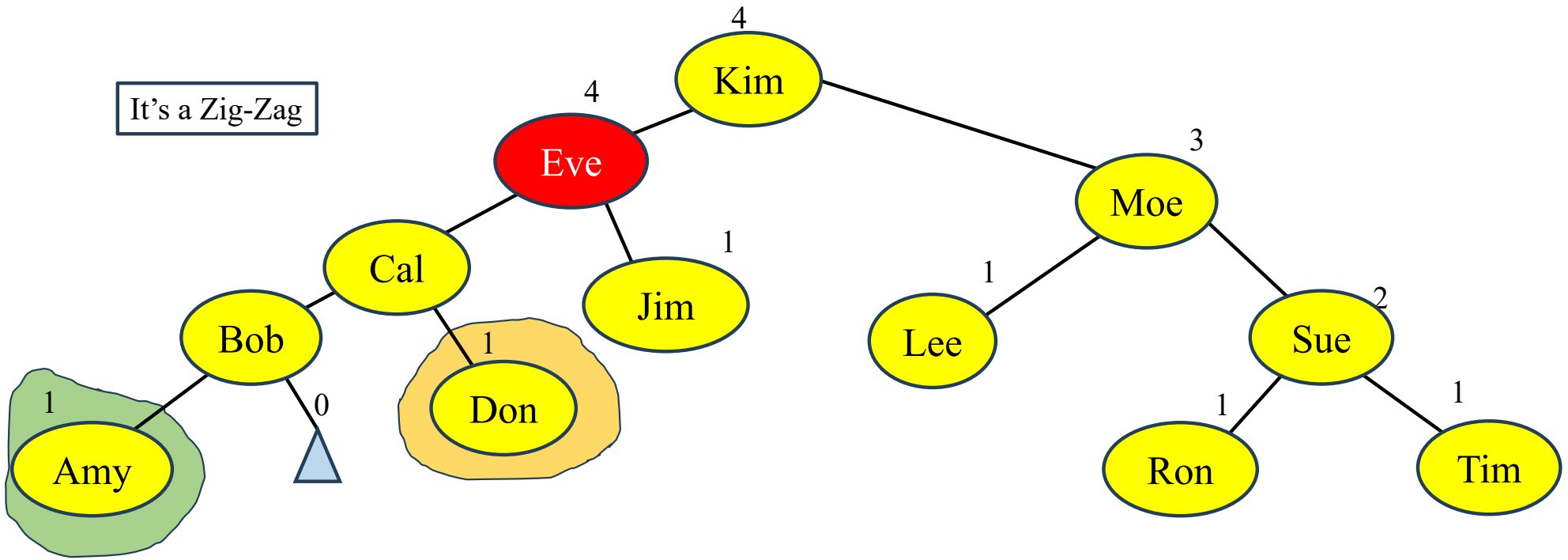
Example:
Insert Don:

Rotate twice about Cal



Example:
Insert Don:

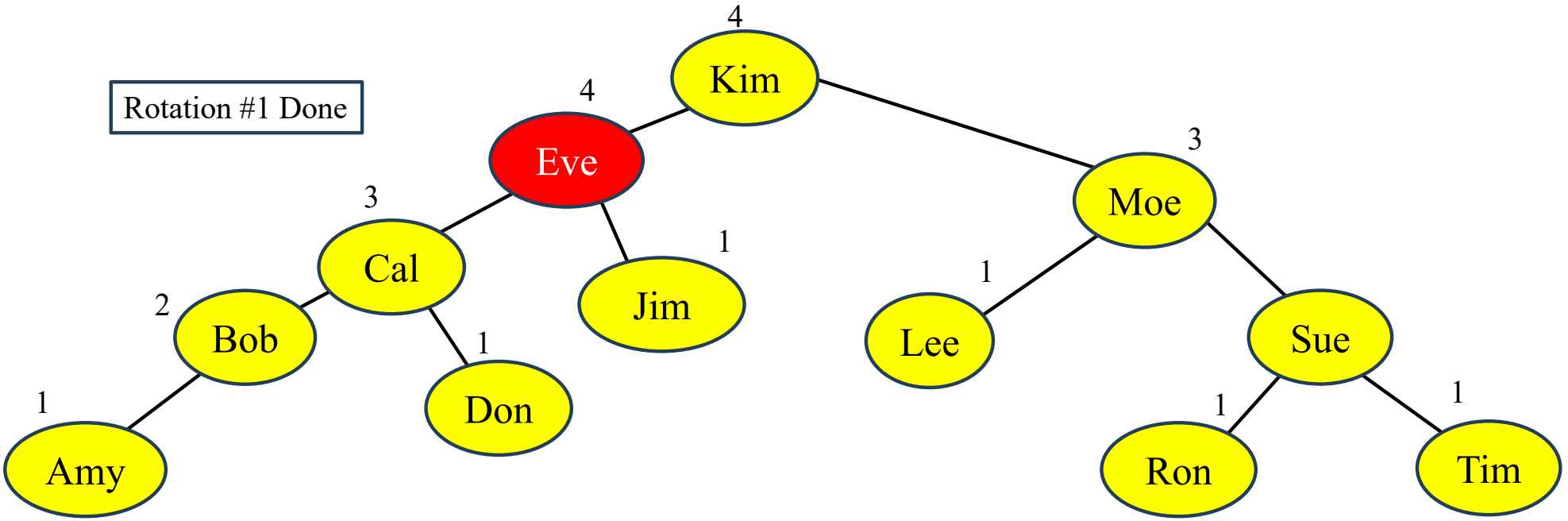
Rotate twice about Cal



Example:
Insert Don:

Rotate twice about Cal

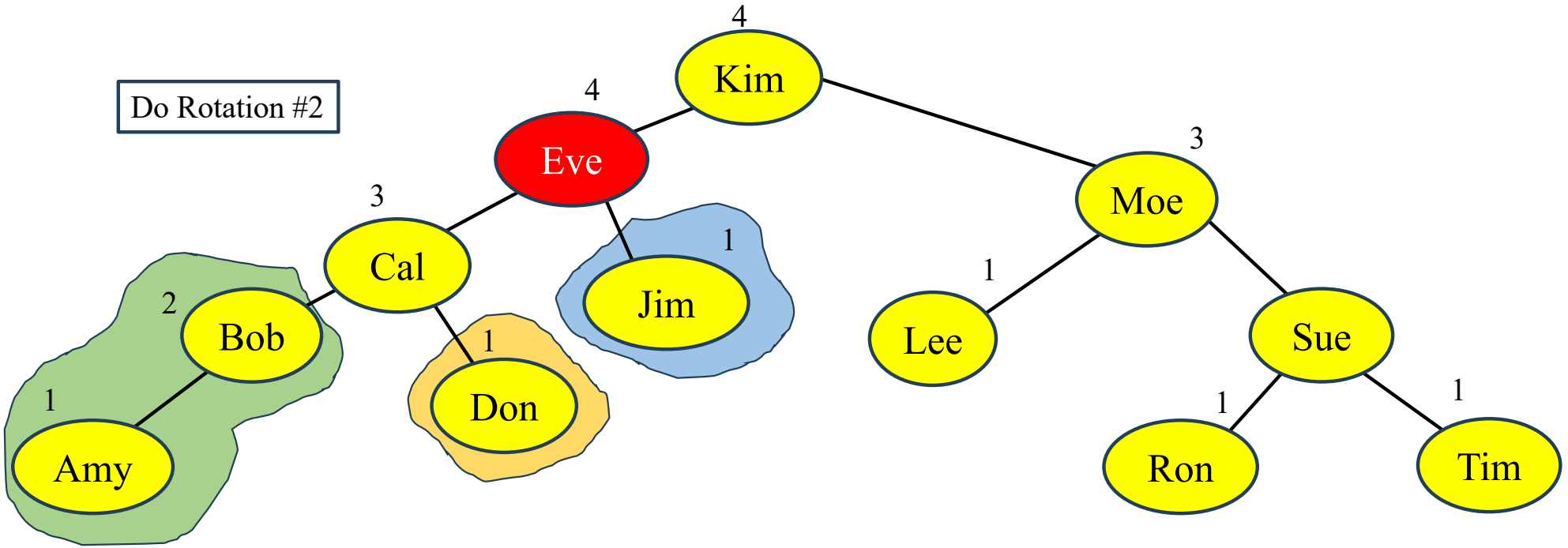
Rotation #1 Done



Example:
Insert Don:

Rotate twice about Cal

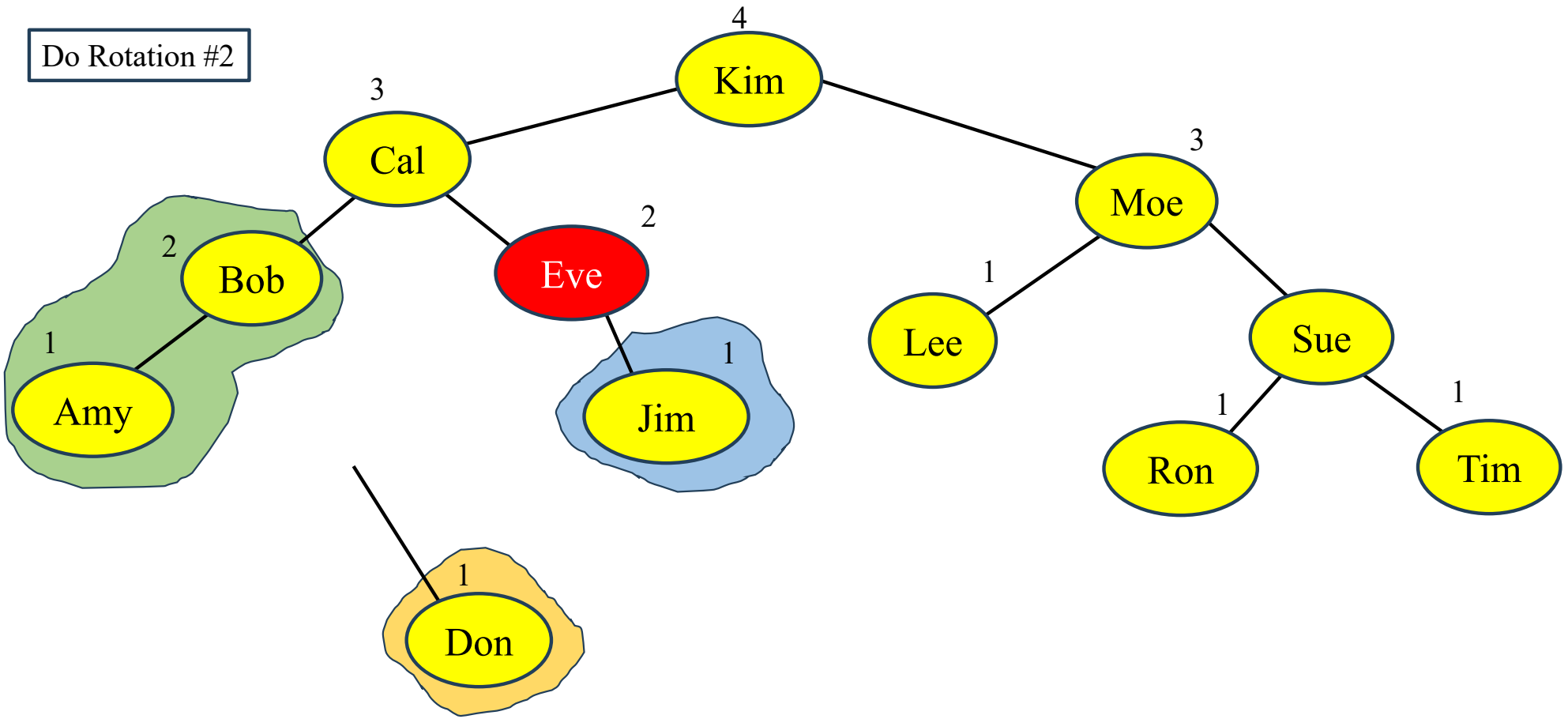
Do Rotation #2



Example:
Insert Don:

Rotate twice about Cal

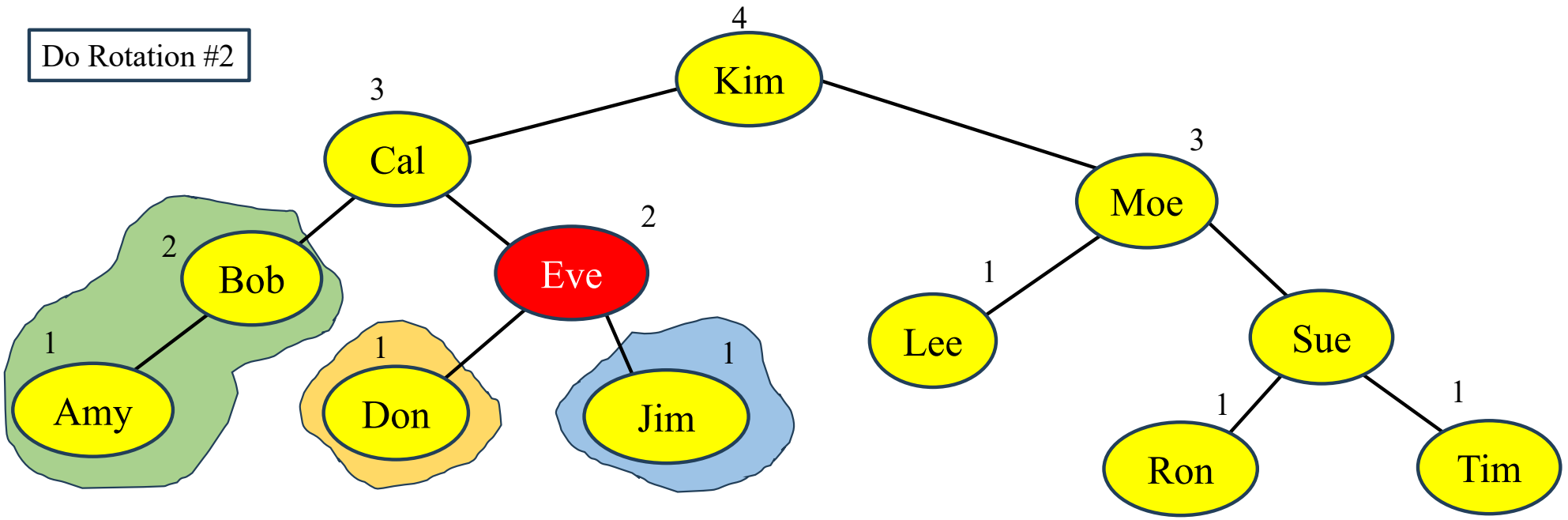
Do Rotation #2



Example:
Insert Don:

Rotate twice about Cal

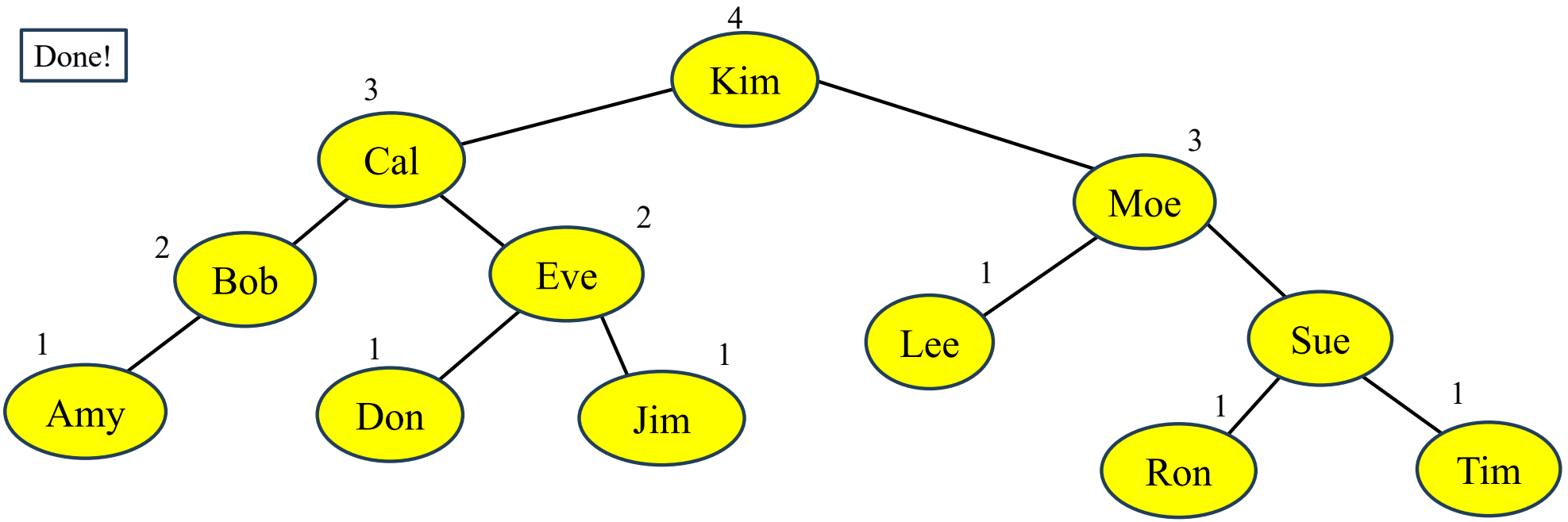
Do Rotation #2



Example:
Insert Don:

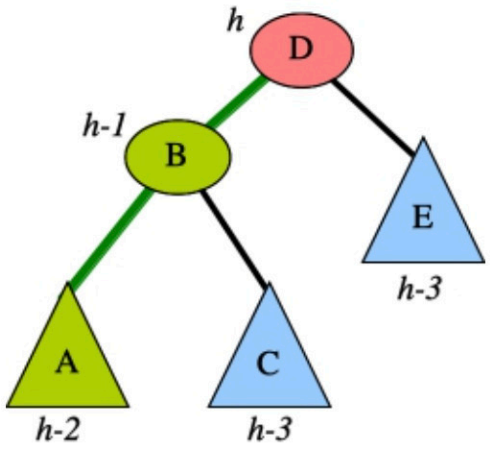
Rotate twice about Cal

Done!

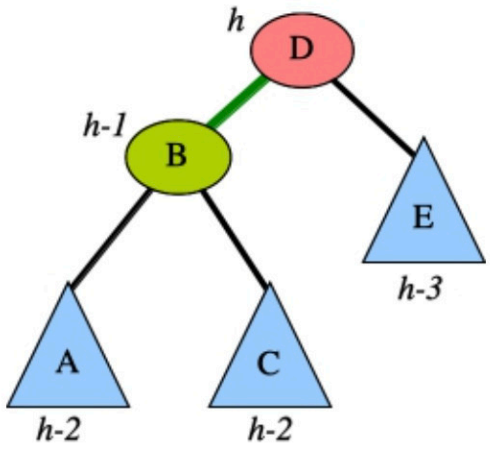


Deletion

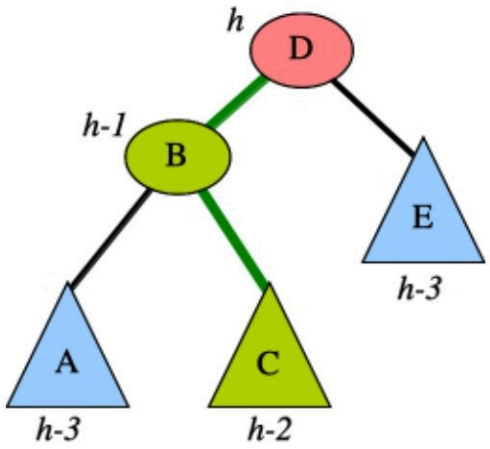
Deletion – can end up with these imbalanced trees.



Zig-Zig -- same as above.

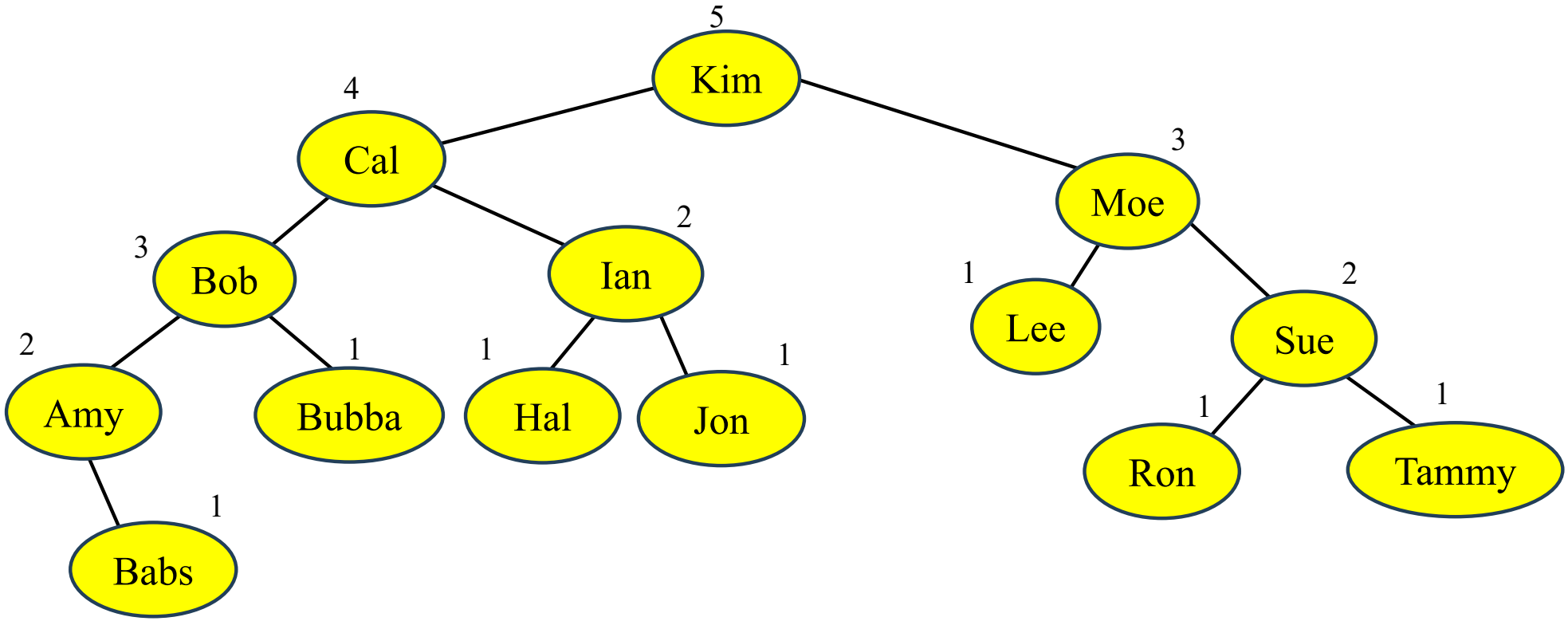


This case only occurs with deletion. We treat it as a Zig-Zig.

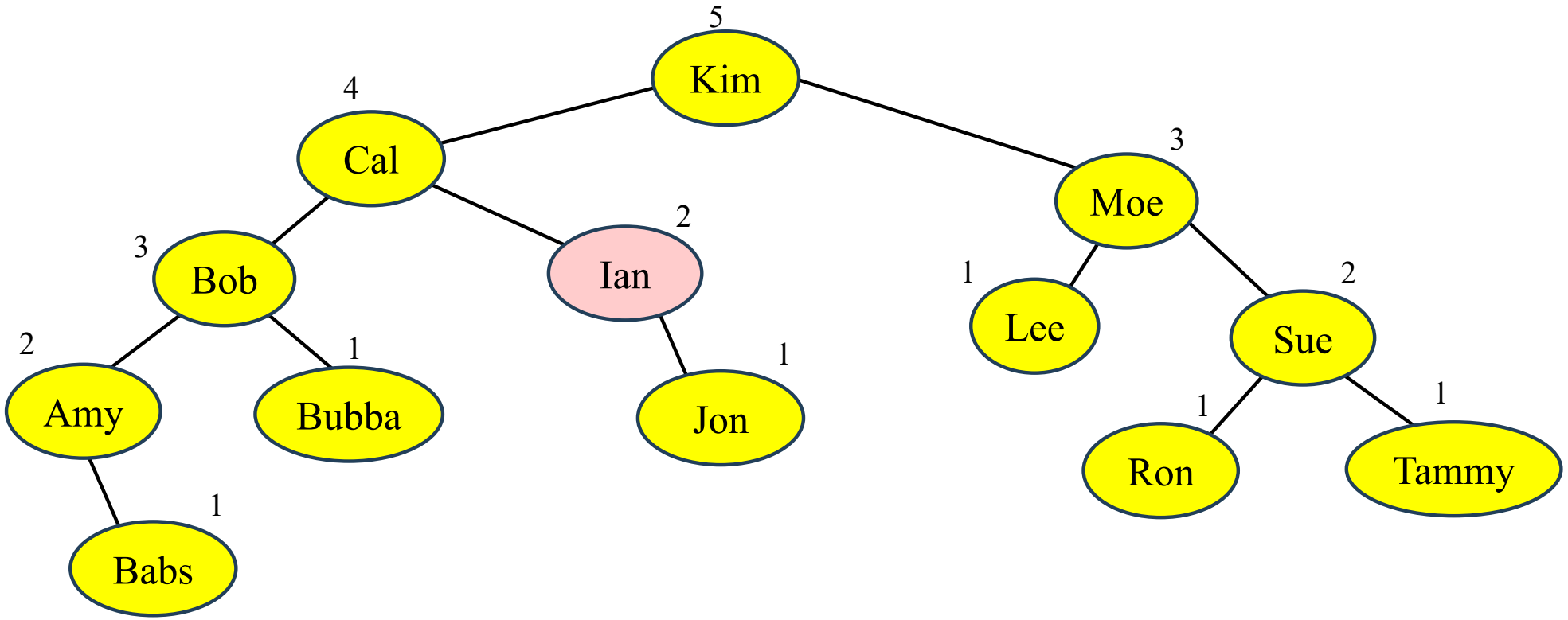


Zig-Zag -- same as above.

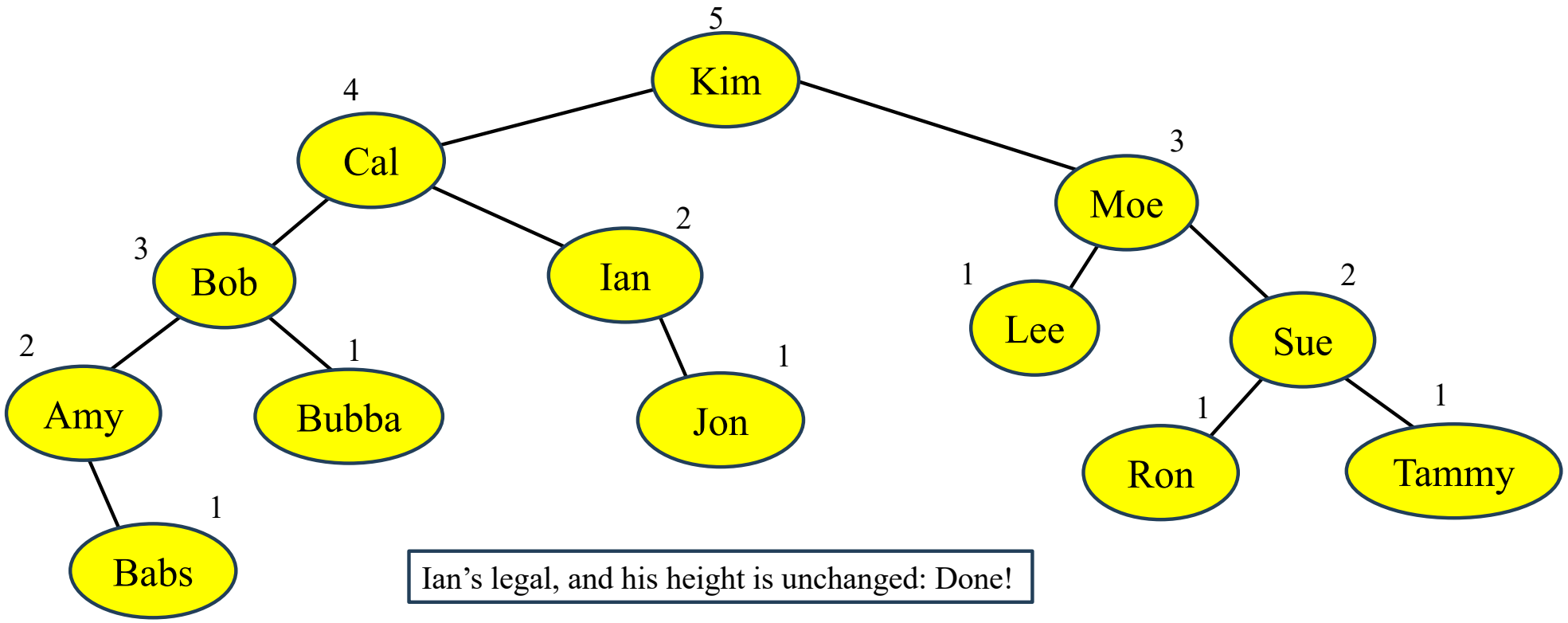
Example:
Delete Hal



Example:
Delete Hal

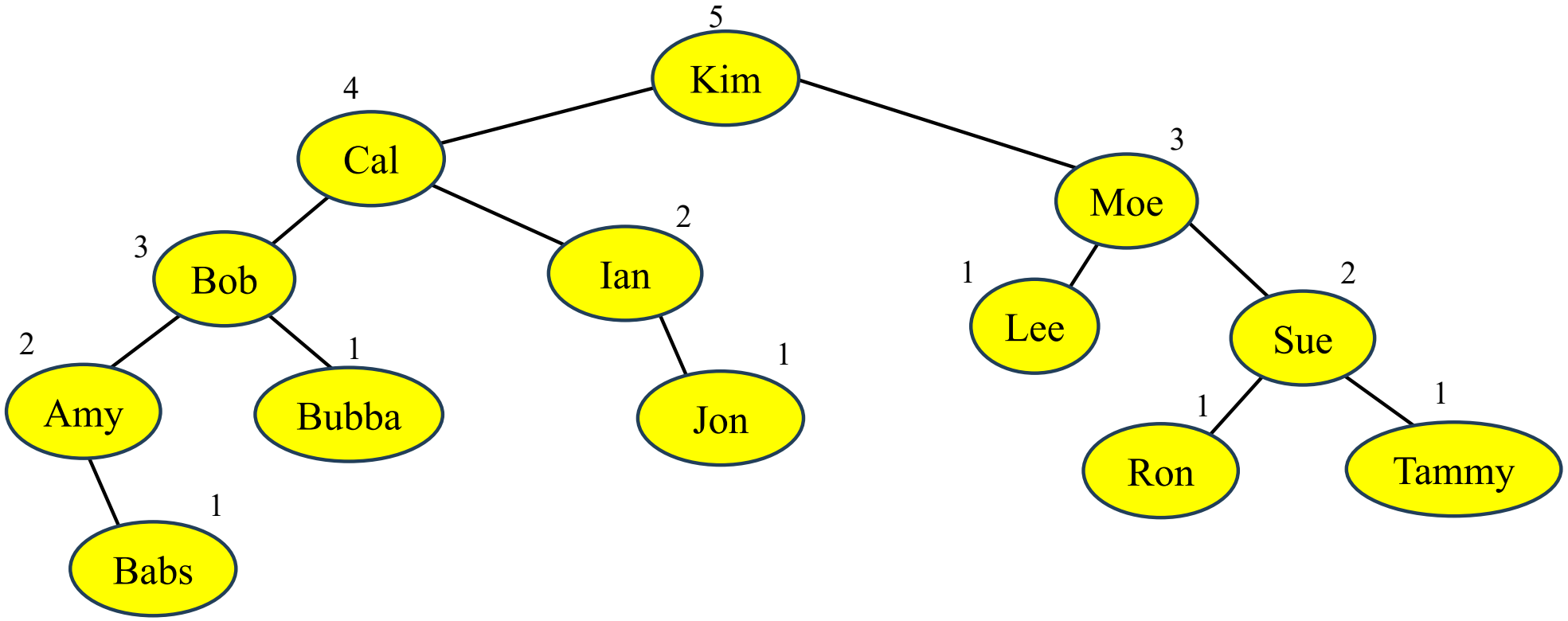


Example:
Delete Hal

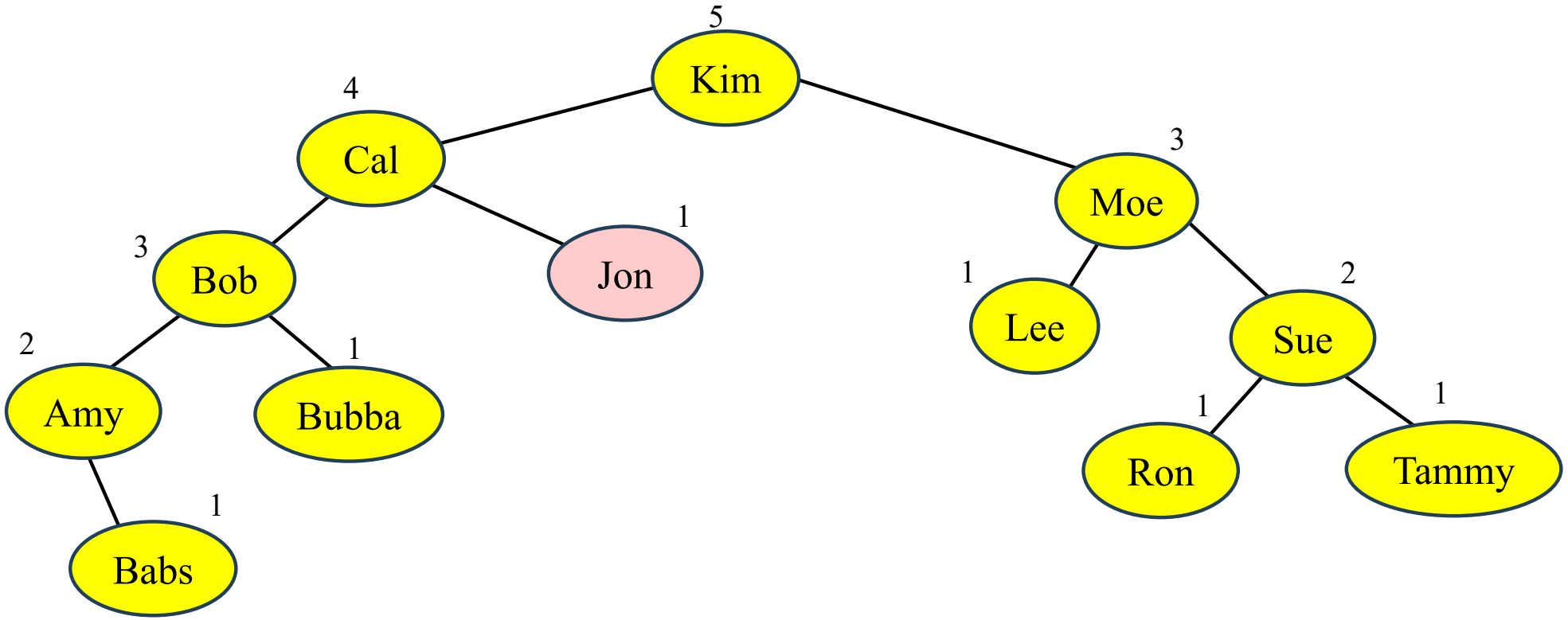


Ian's legal, and his height is unchanged: Done!

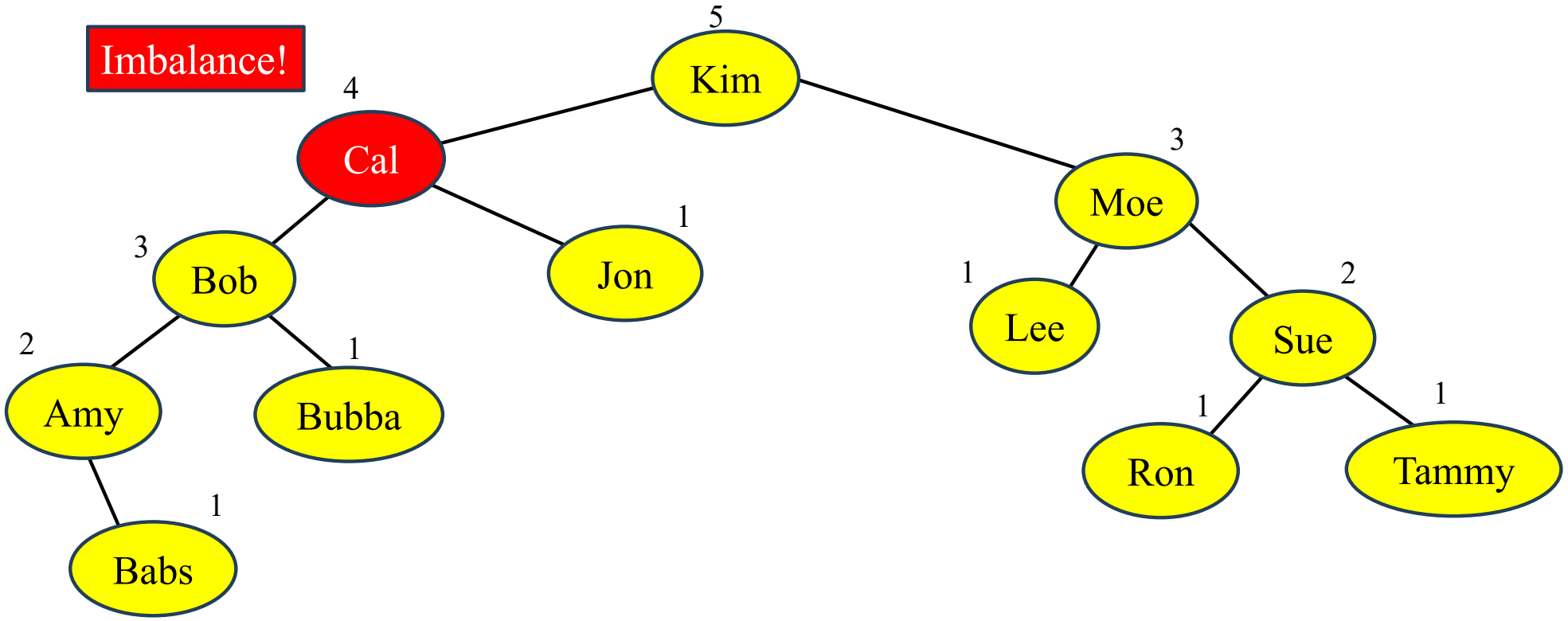
Example:
Delete Ian



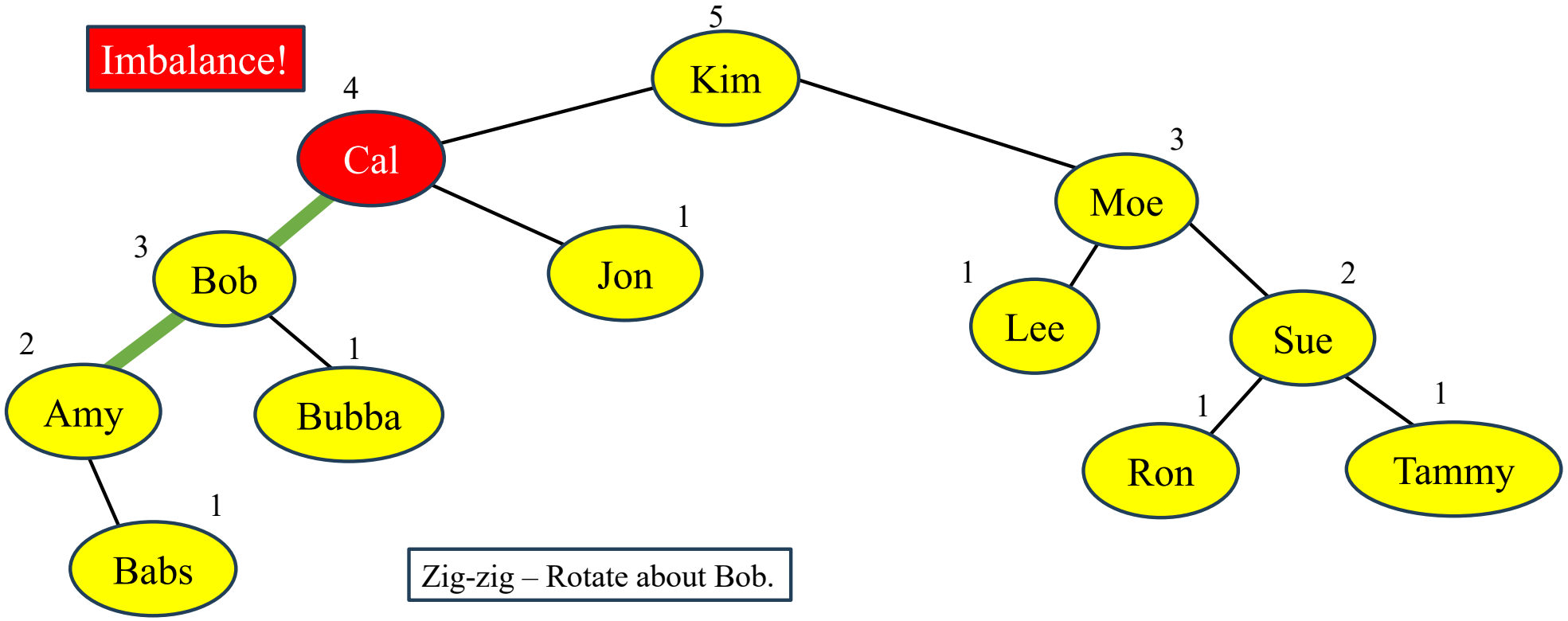
Example:
Delete Ian



Example:
Delete Ian

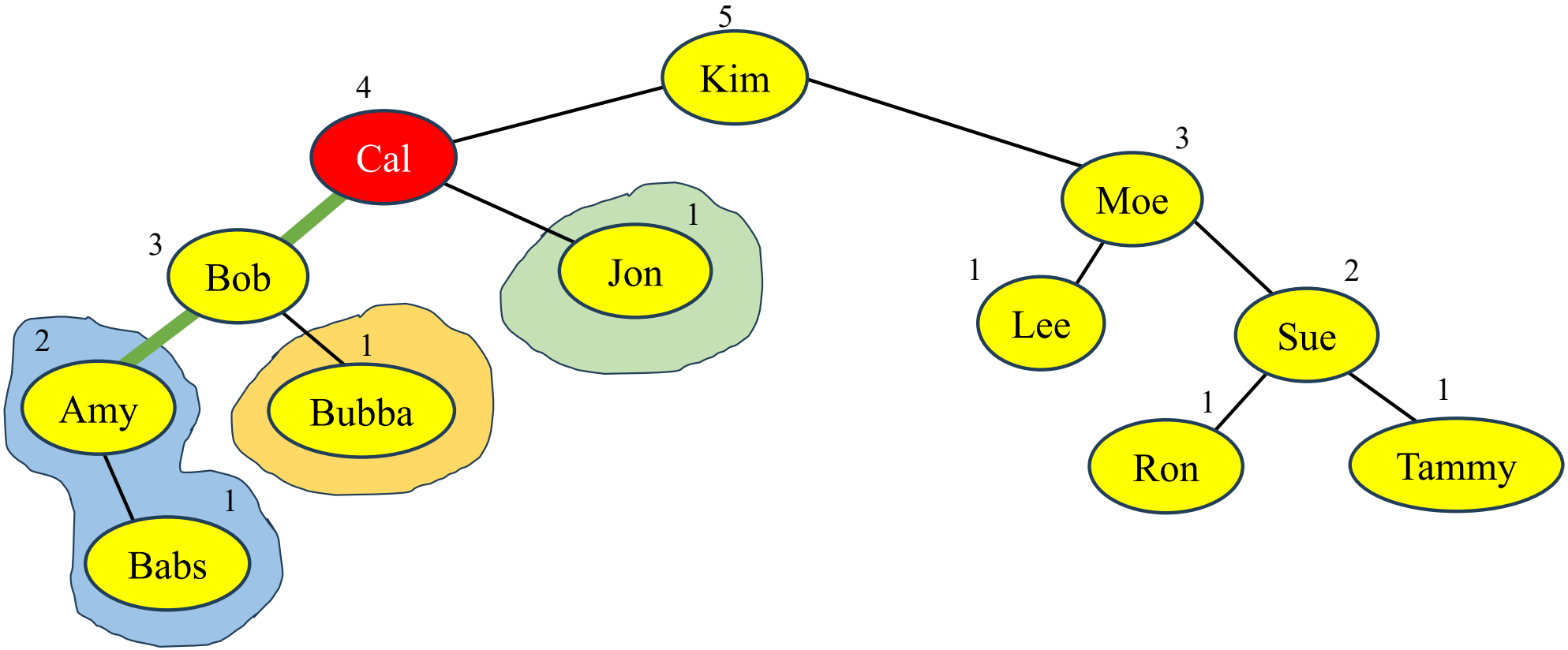


Example:
Delete Ian



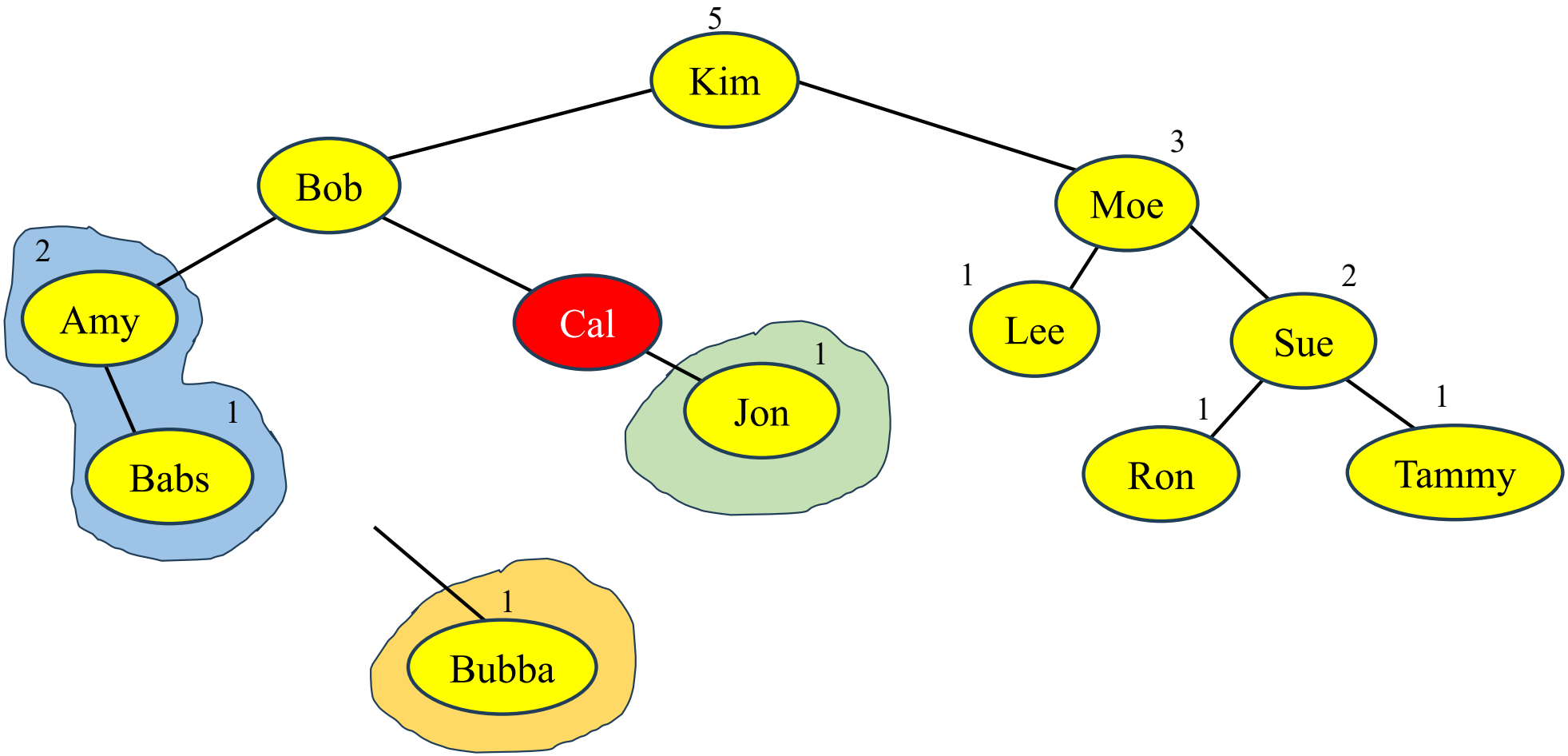
Example:
Delete Ian

Zig-zig – Rotate about Bob.



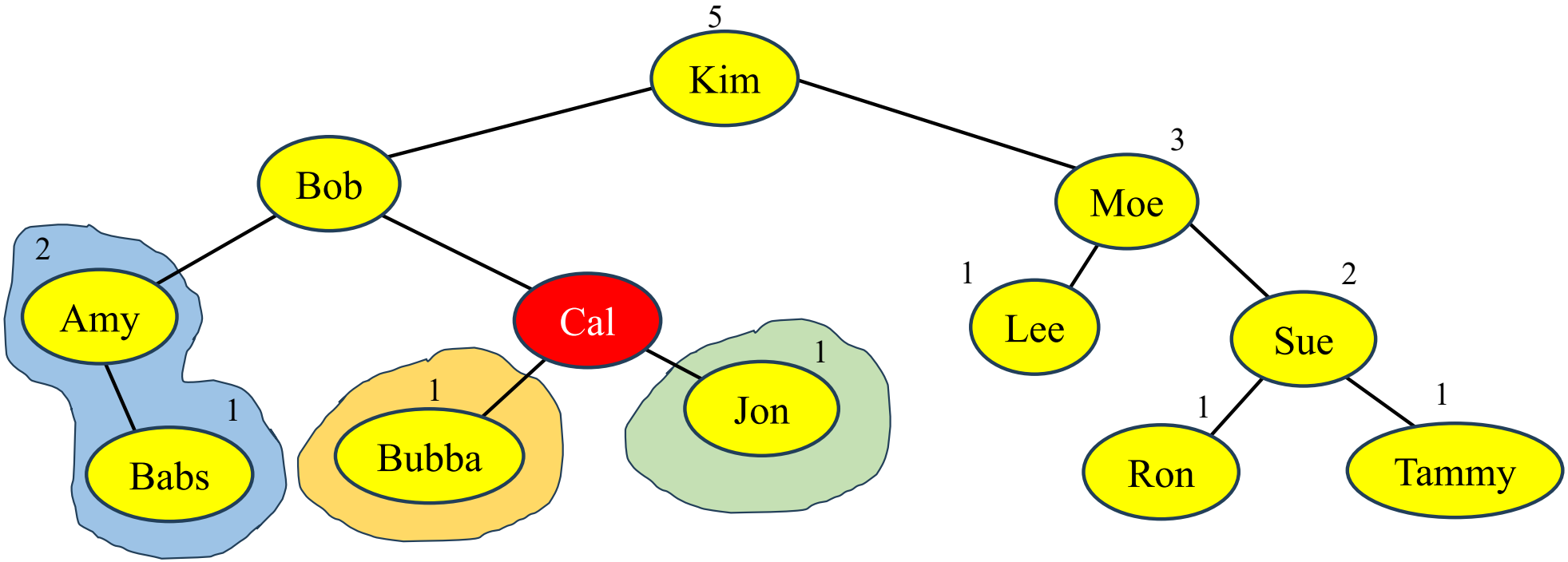
Example:
Delete Ian

Zig-zig – Rotate about Bob.



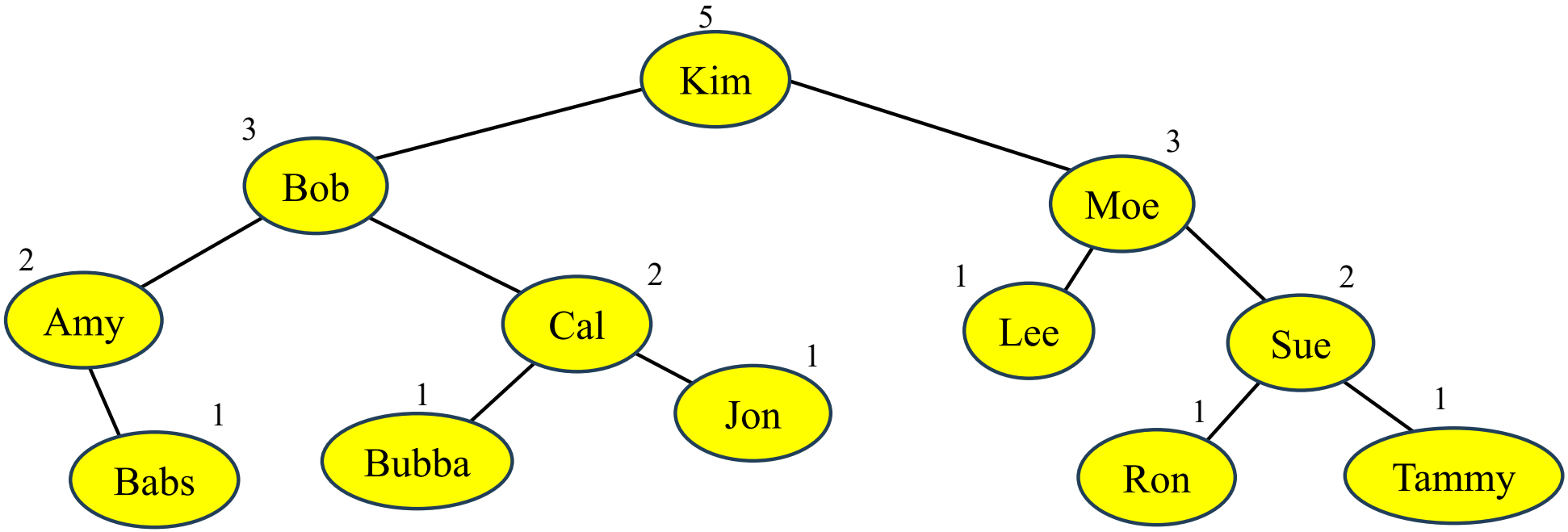
Example:
Delete Ian

Zig-zig – Rotate about Bob.



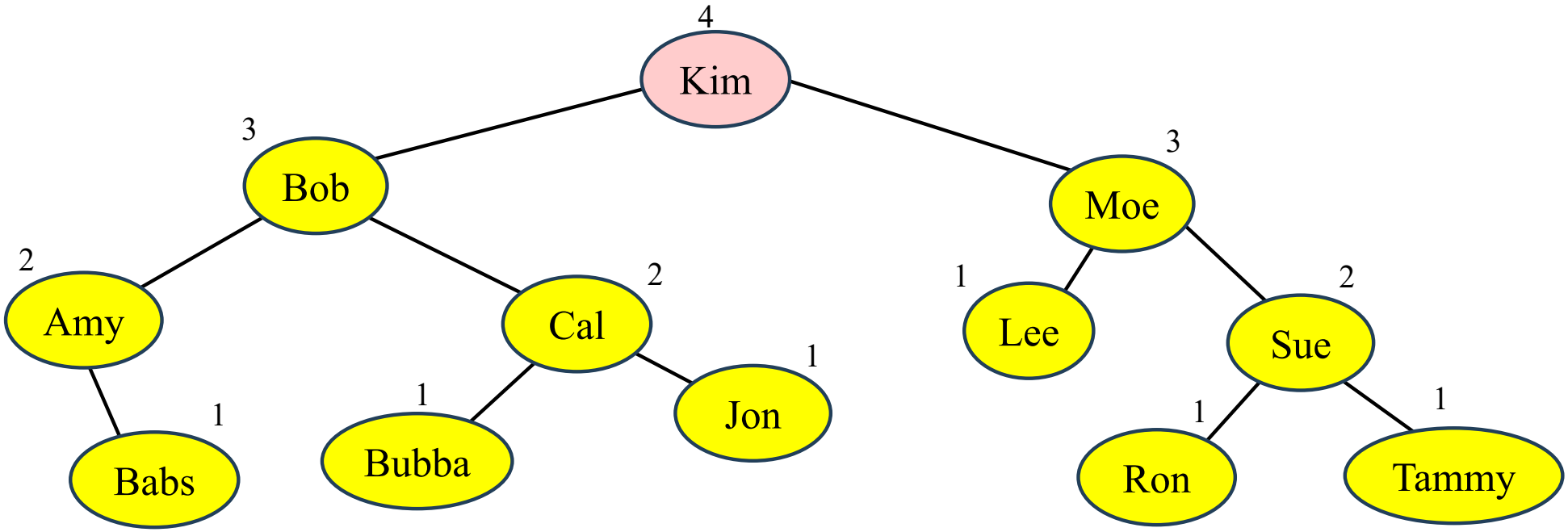
Example:
Delete Ian

Done – But we still need to check heights to the root:

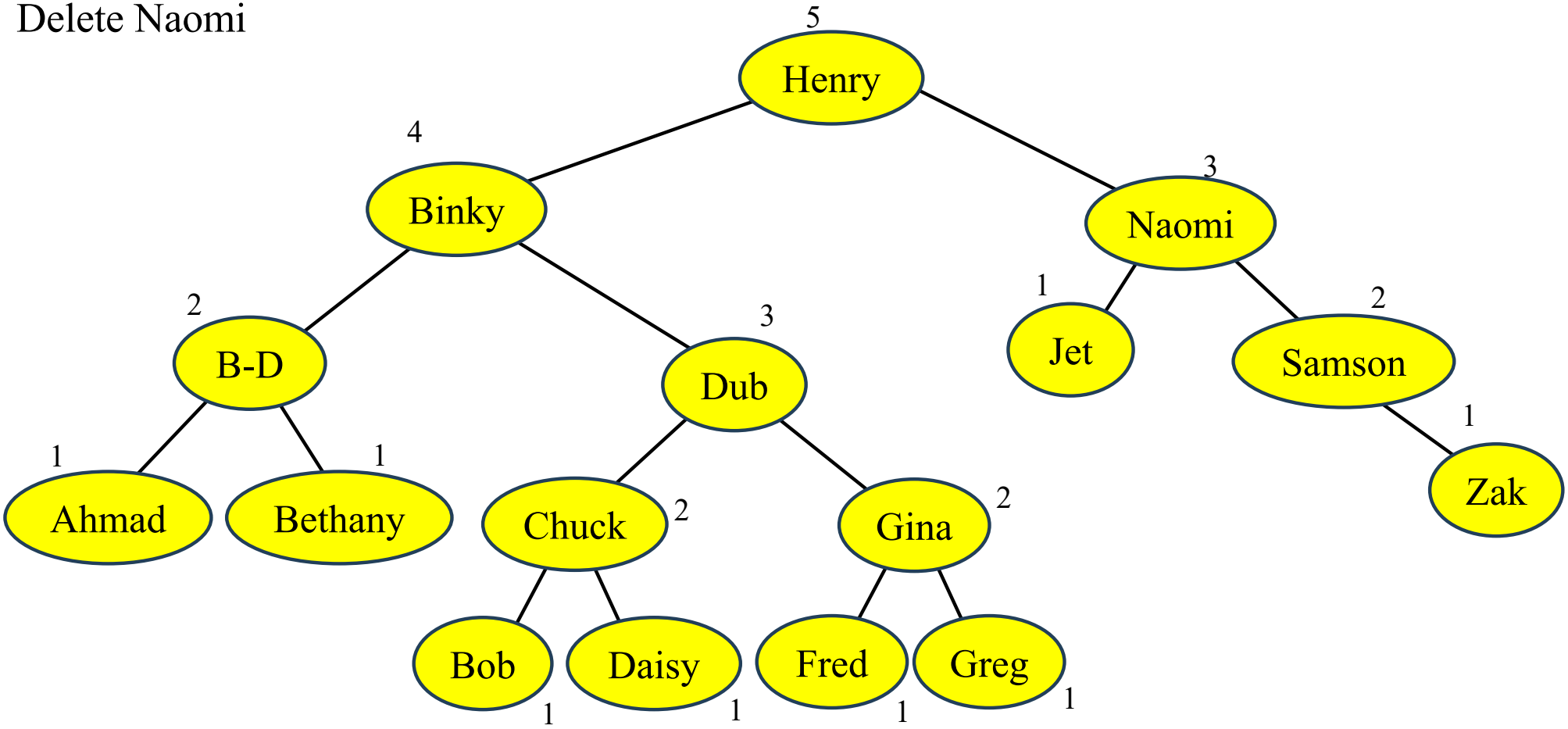


Example:
Delete Ian

Done – But we still need to check heights to the root:

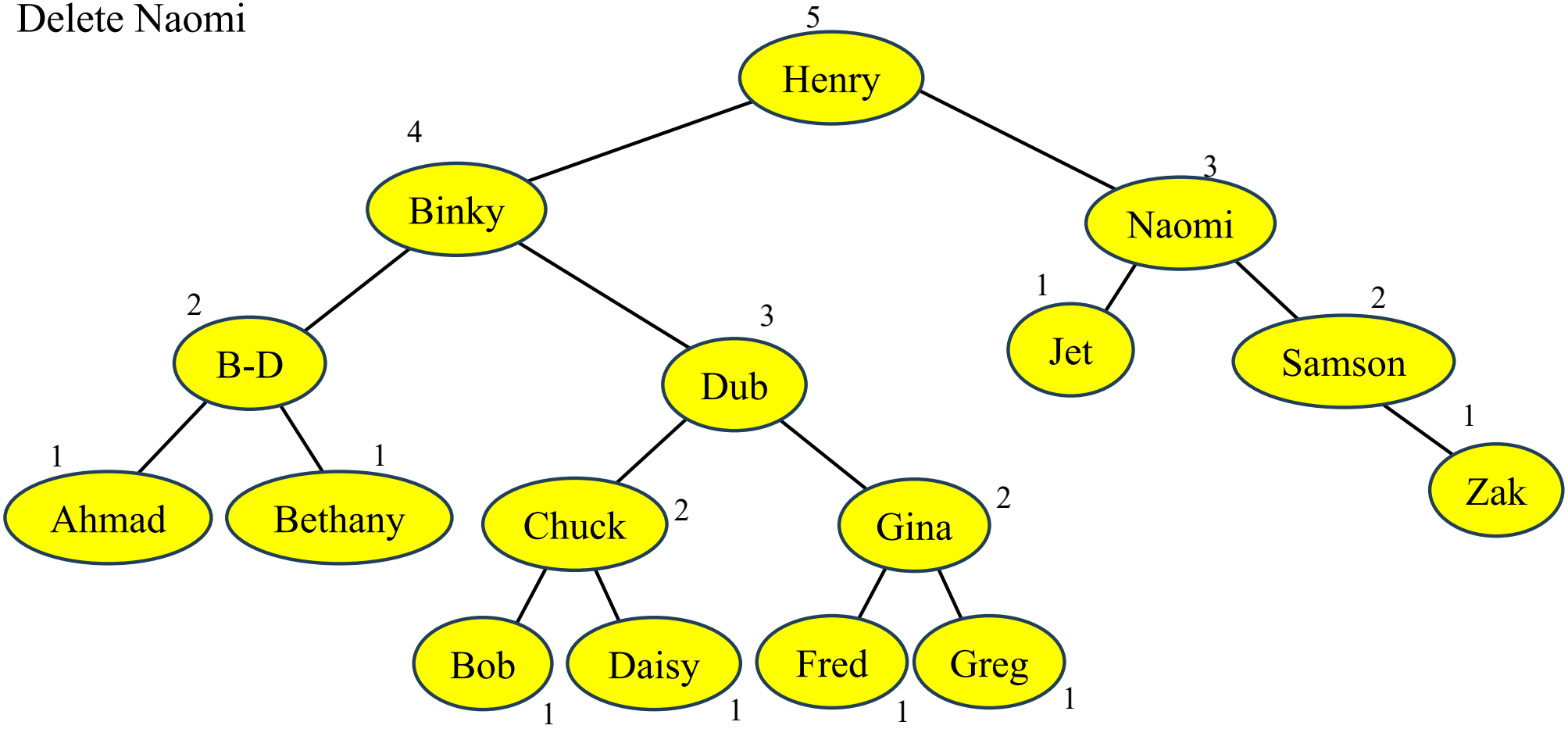


Final Example:
Delete Naomi



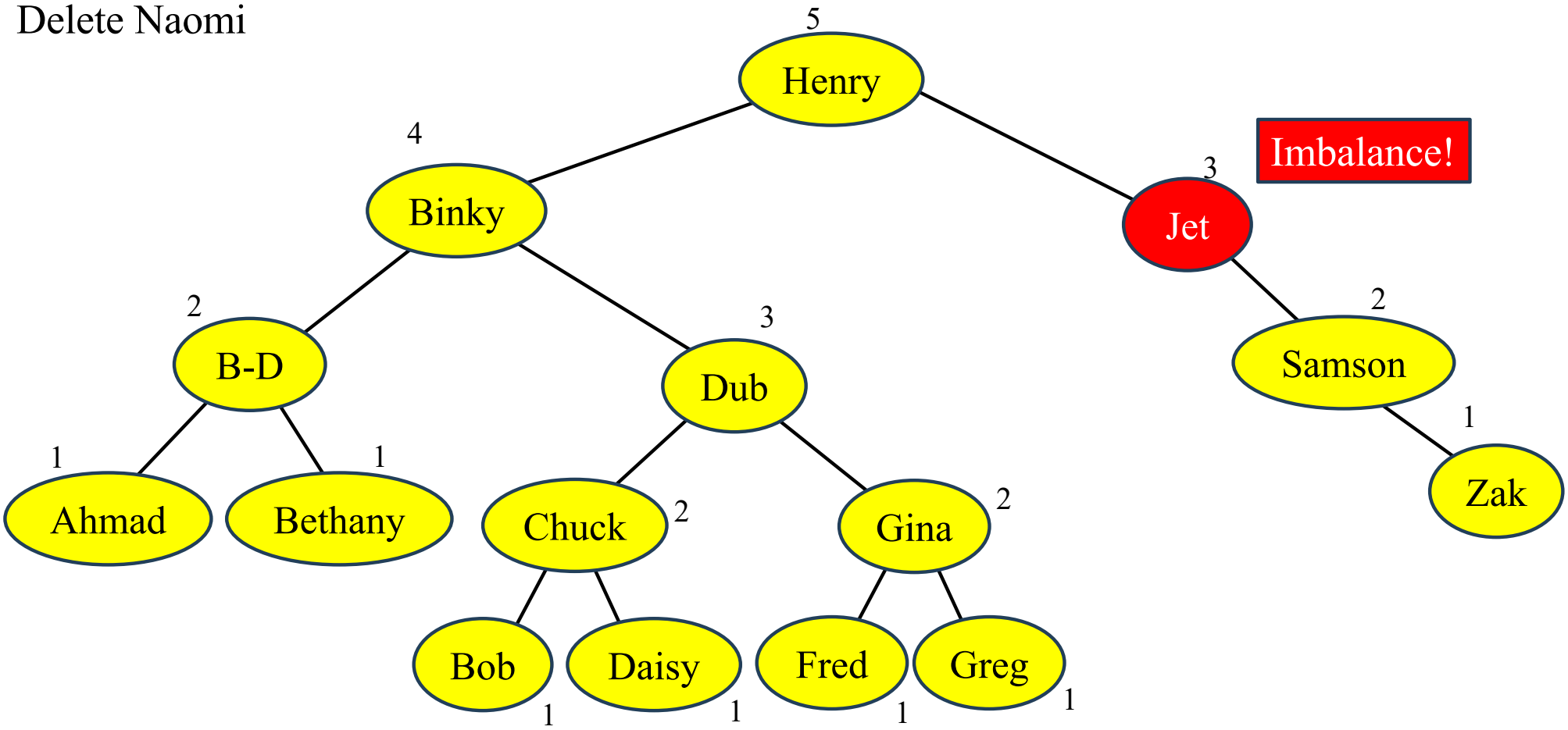
Final Example:
Delete Naomi

Delete Jet and replace Naomi with Jet:



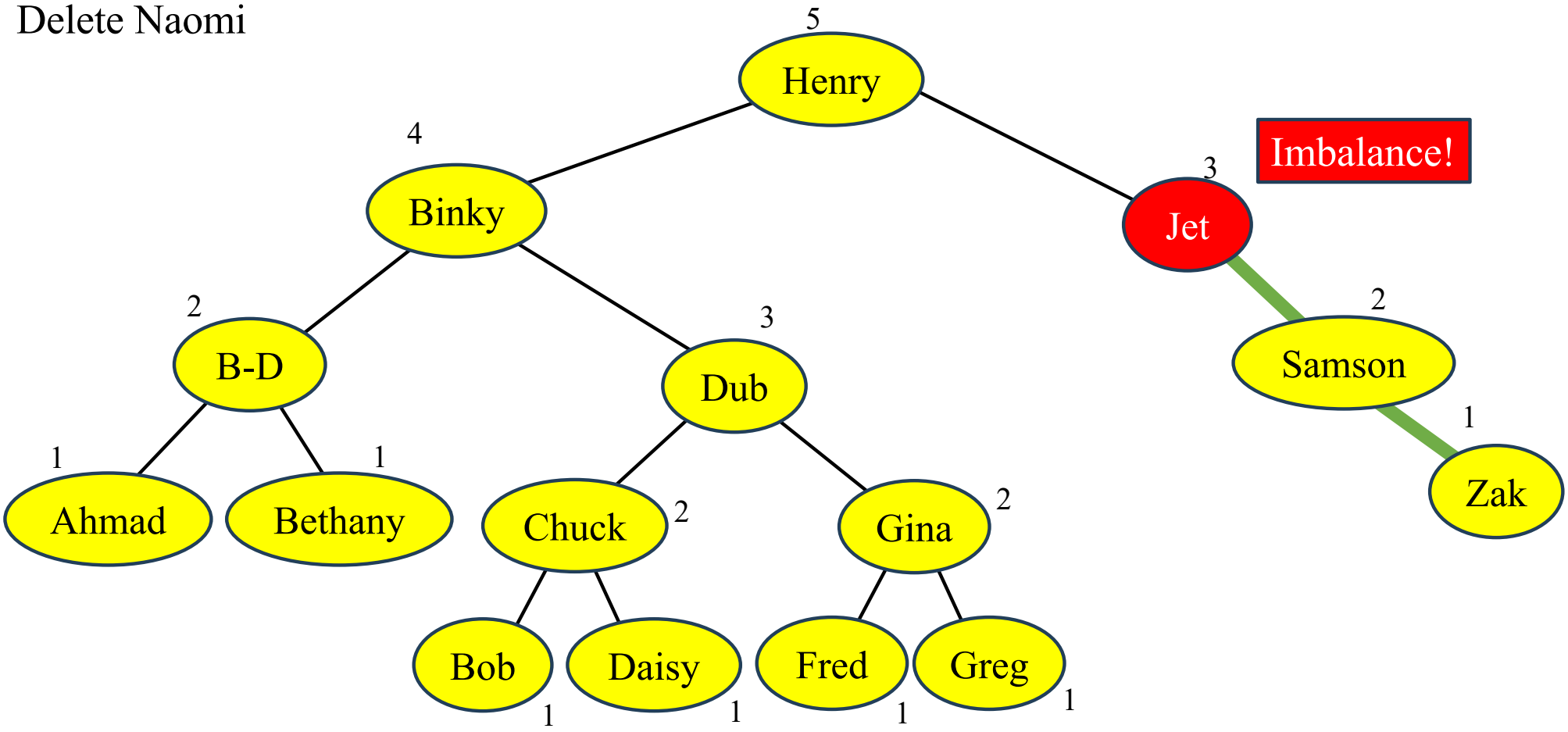
Final Example:
Delete Naomi

Delete Jet and replace Naomi with Jet:



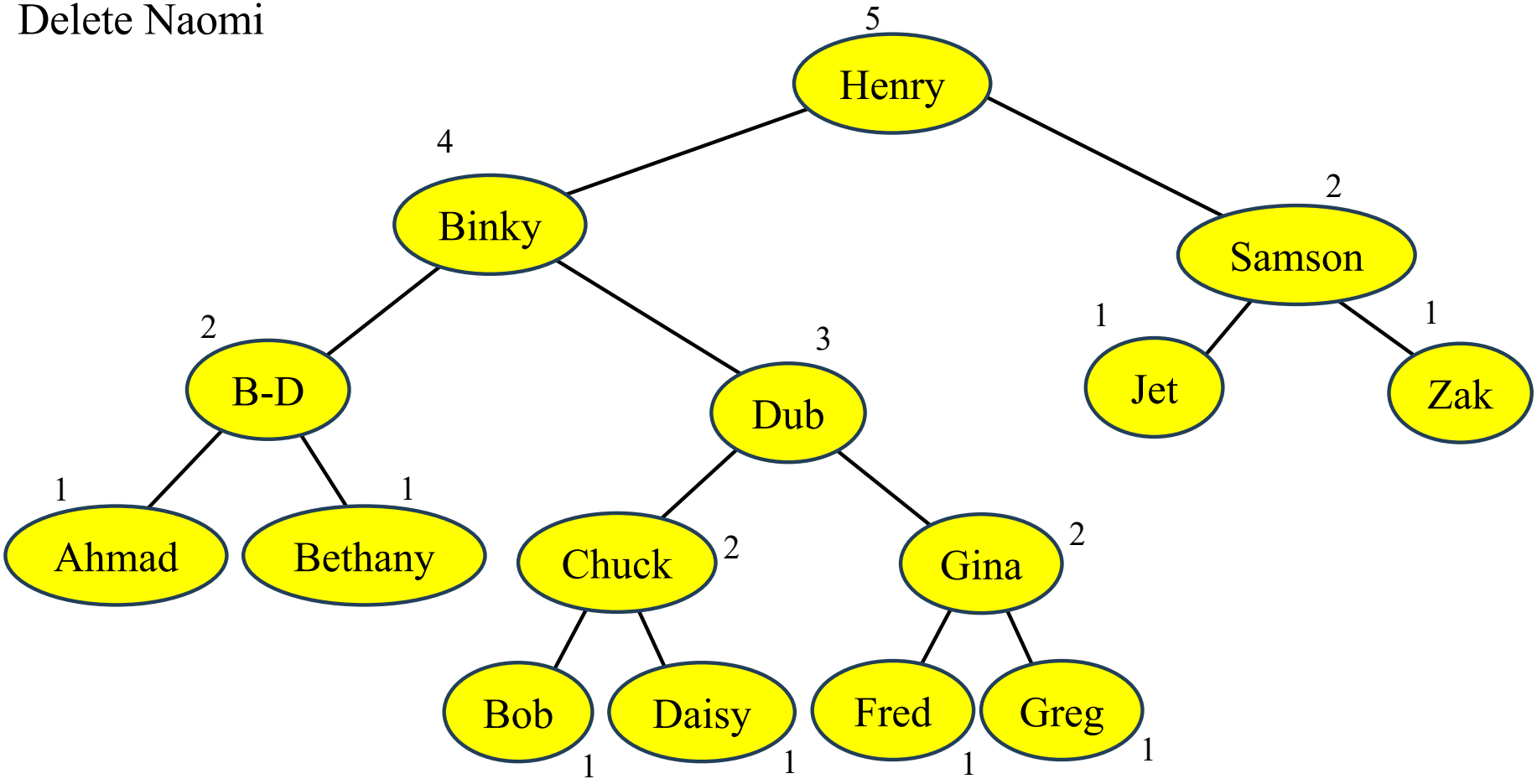
Final Example:
Delete Naomi

It's a Zig-Zig, so rotate about Samson:



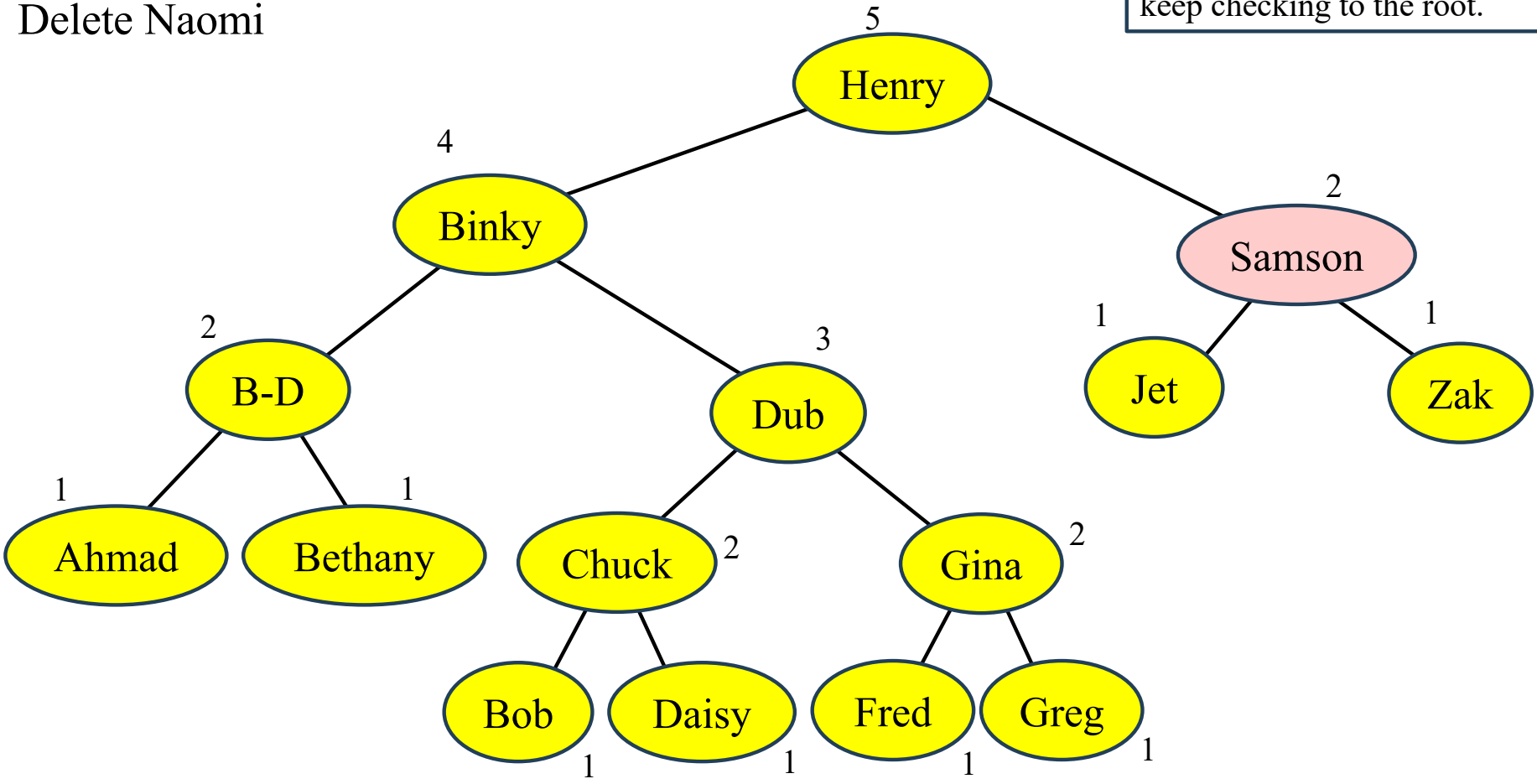
Final Example:
Delete Naomi

It's a Zig-Zig, so rotate about Samson:

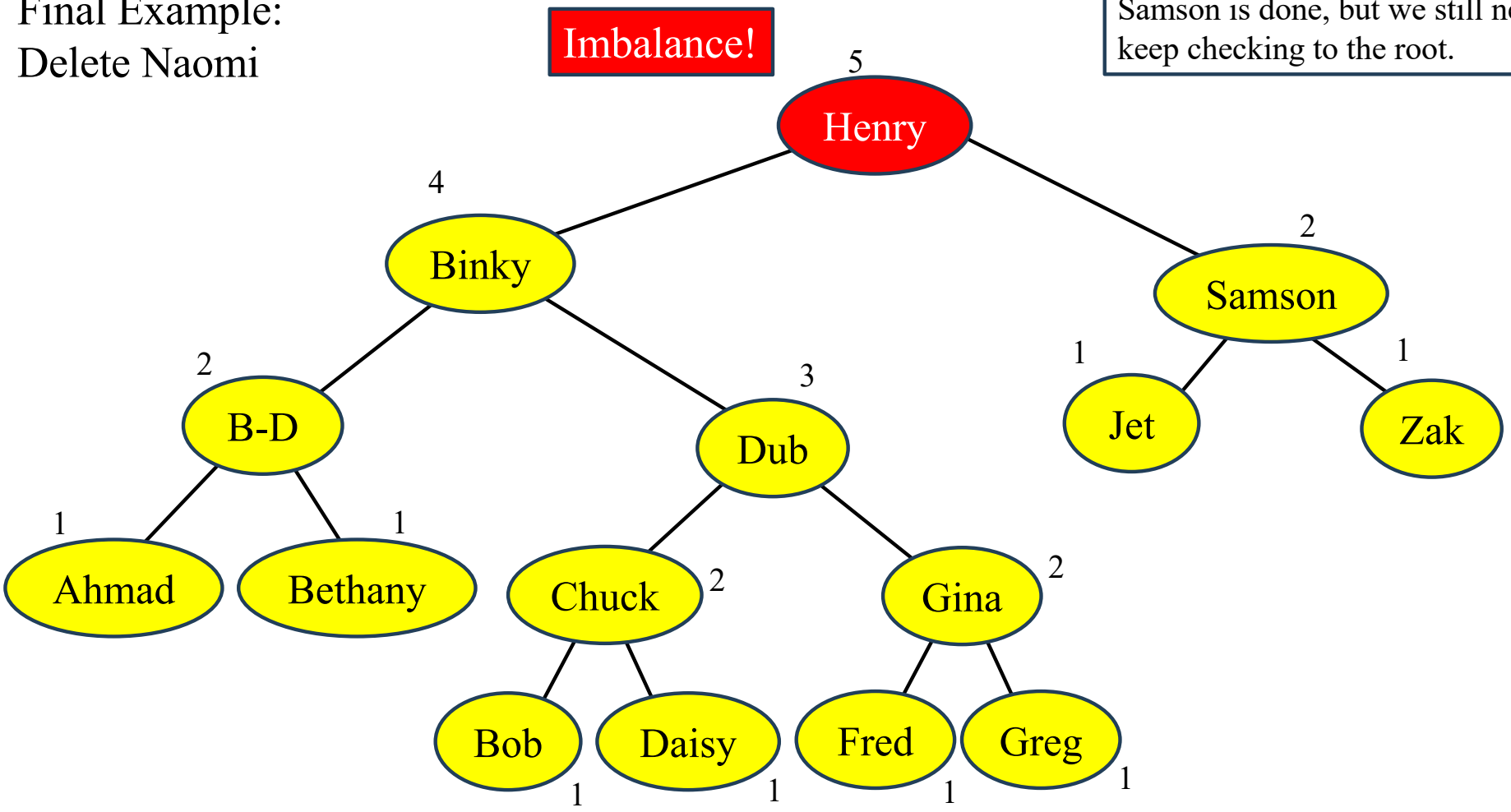


Final Example:
Delete Naomi

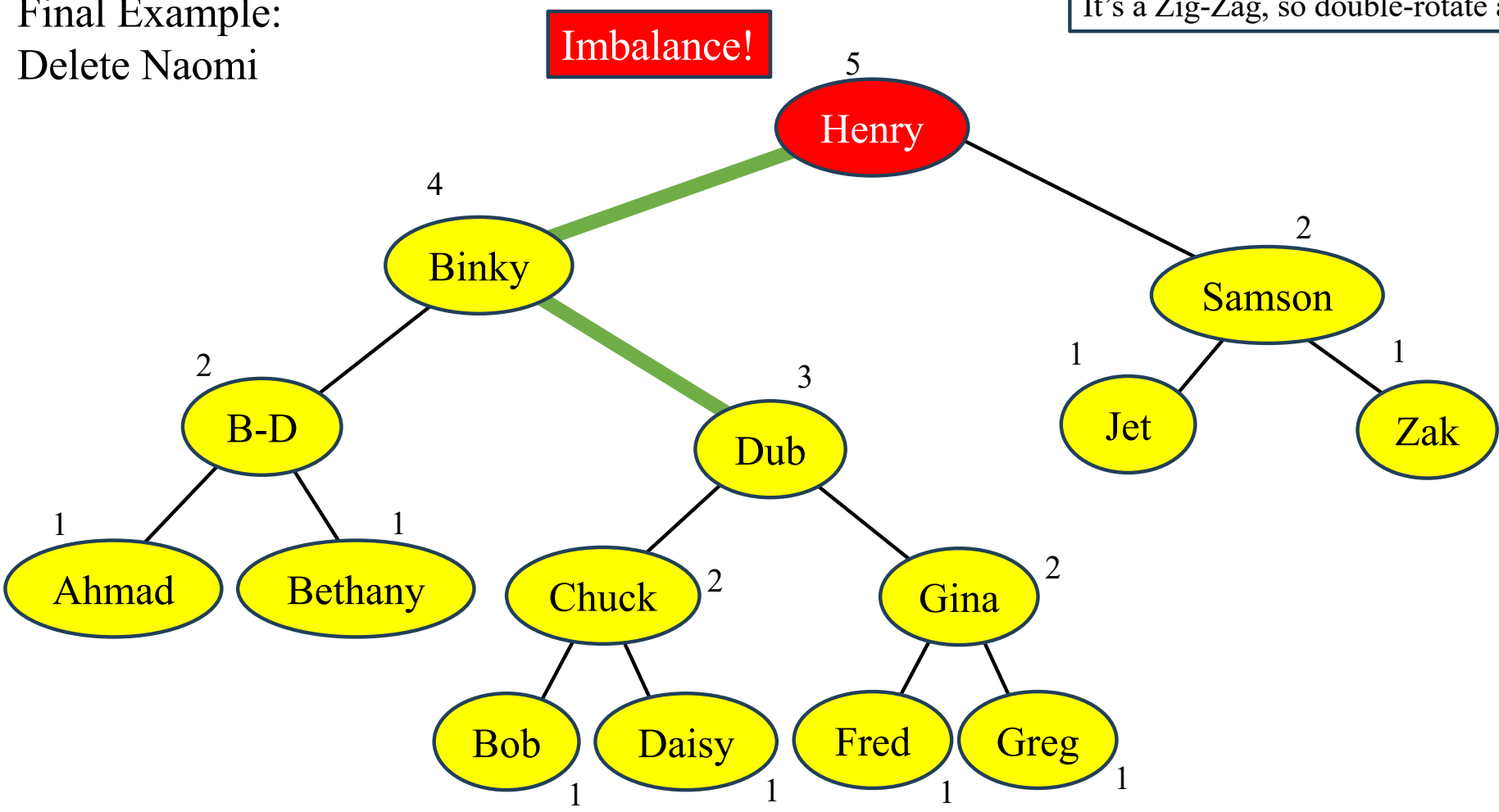
Samson is done, but we still need to keep checking to the root.



Final Example:
Delete Naomi

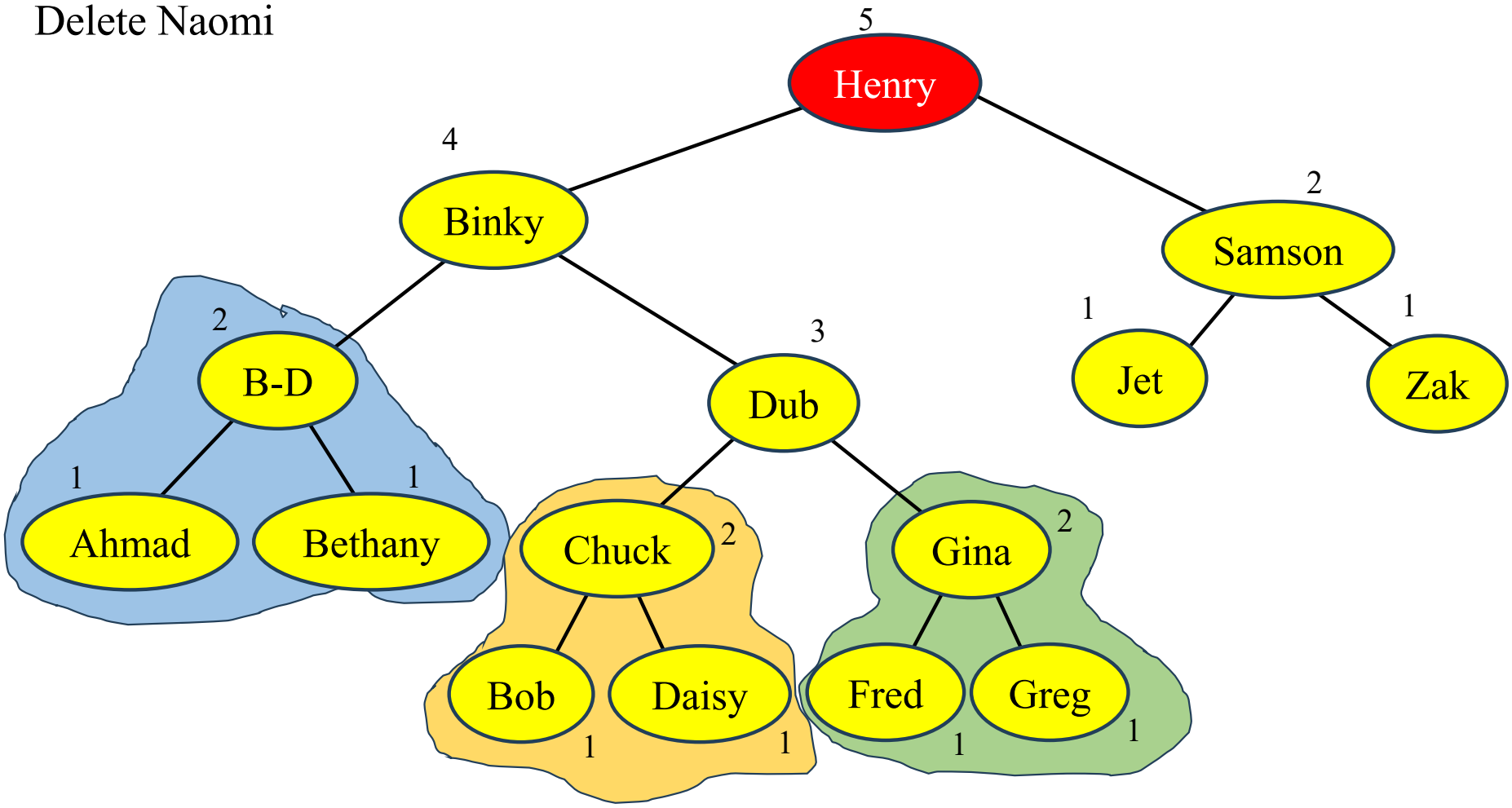


Final Example:
Delete Naomi



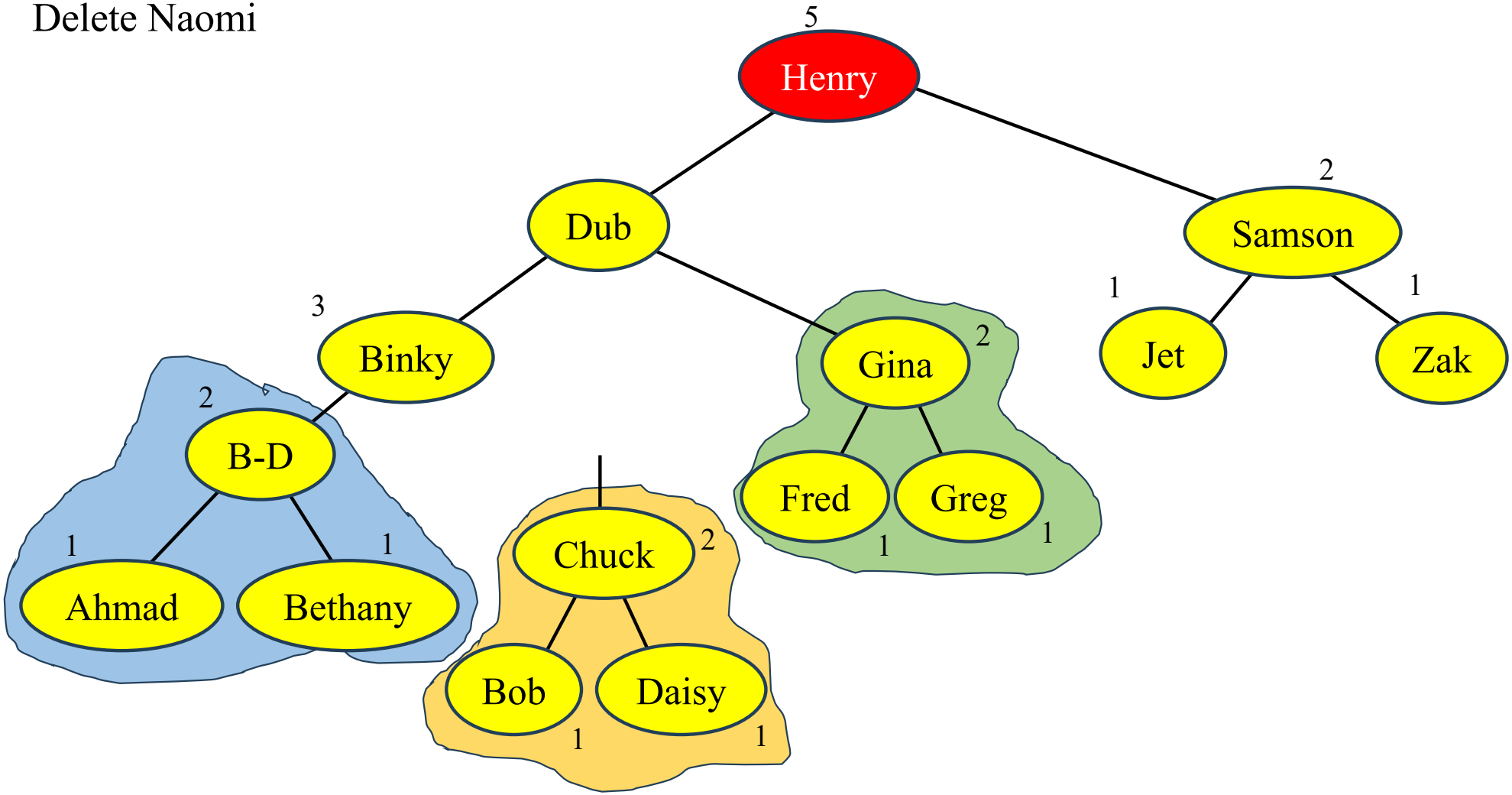
Final Example:
Delete Naomi

It's a Zig-Zag, so double-rotate about Dub.



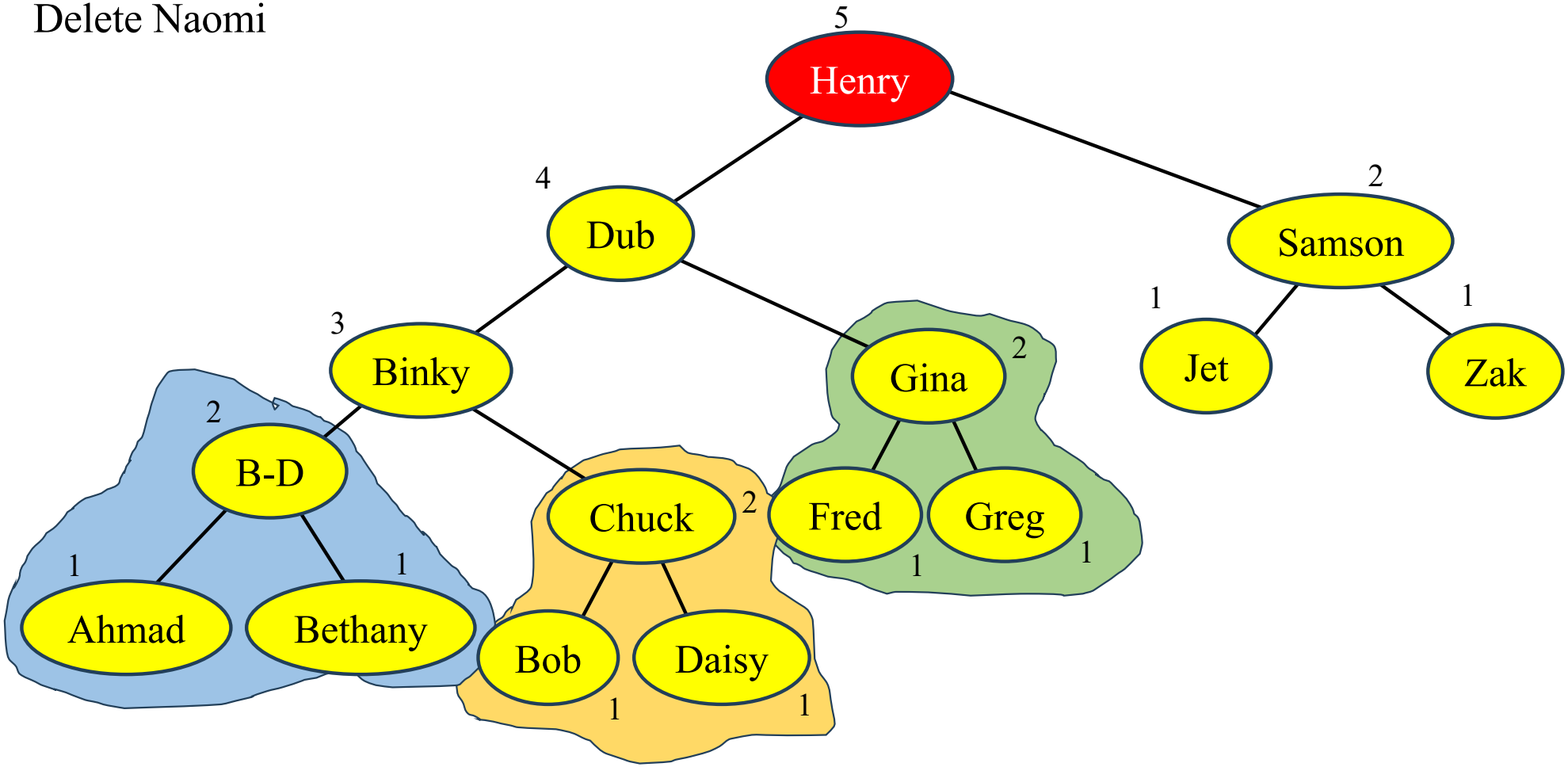
Final Example:
Delete Naomi

It's a Zig-Zag, so double-rotate about Dub.



Final Example:
Delete Naomi

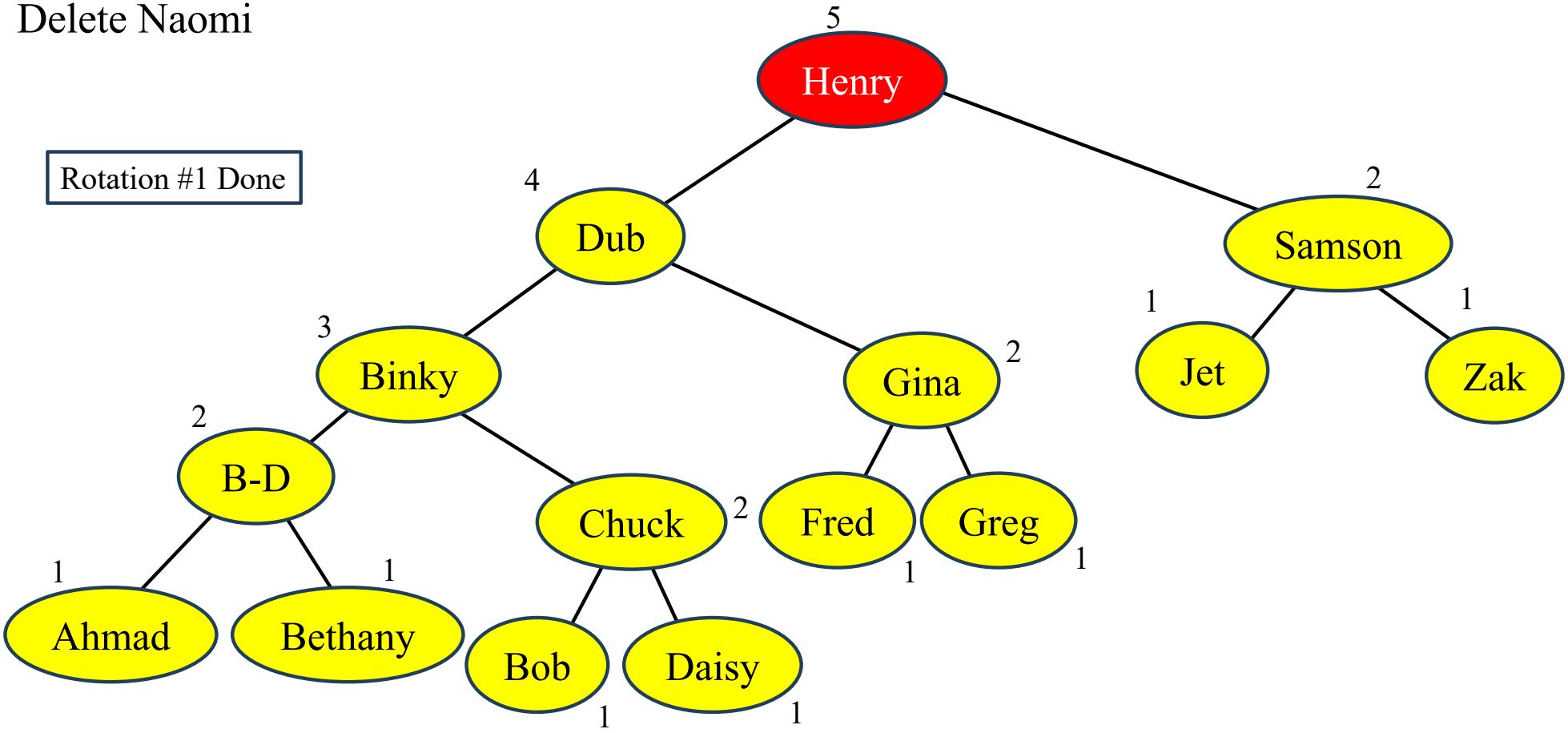
It's a Zig-Zag, so double-rotate about Dub.



Final Example:
Delete Naomi

It's a Zig-Zag, so double-rotate about Dub.

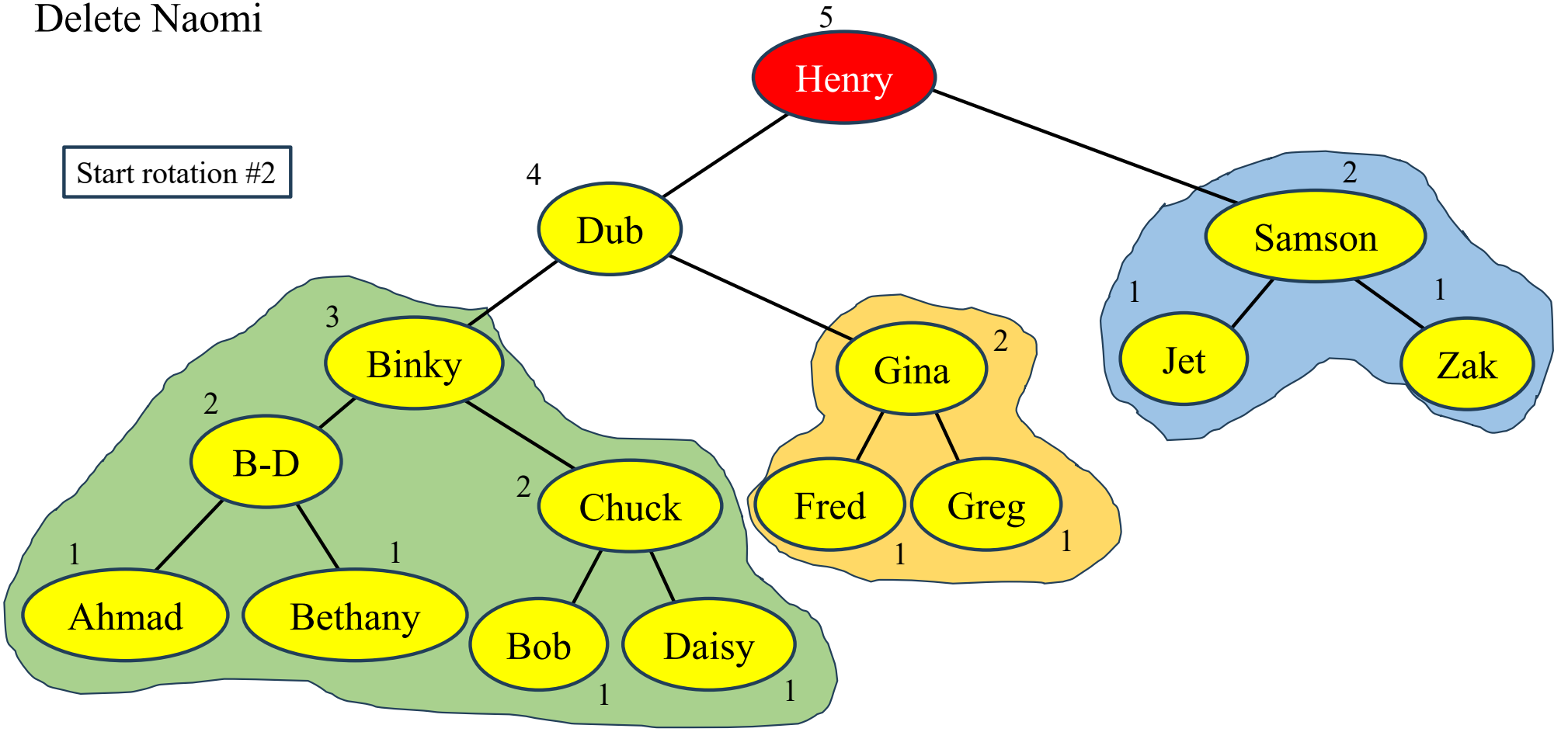
Rotation #1 Done



Final Example:
Delete Naomi

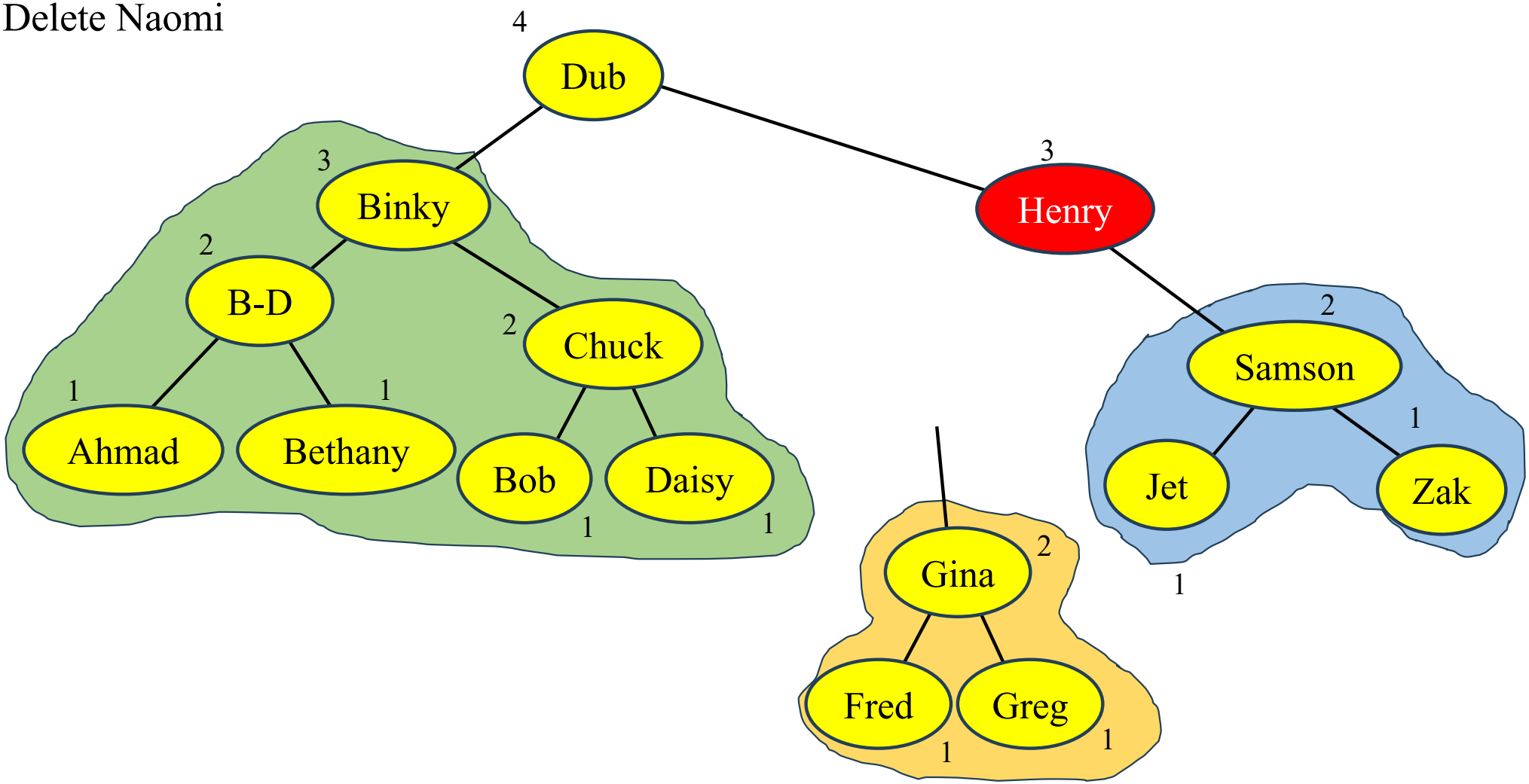
It's a Zig-Zag, so double-rotate about Dub.

Start rotation #2



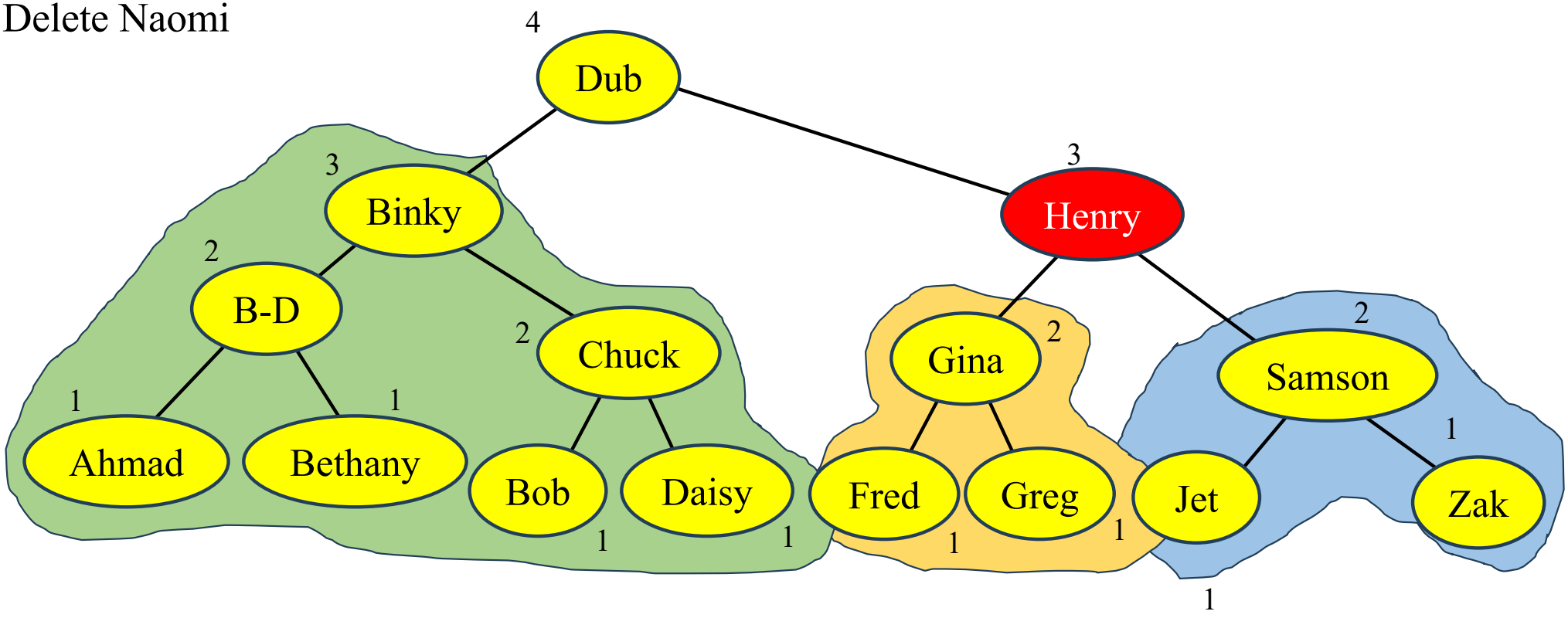
Final Example:
Delete Naomi

It's a Zig-Zag, so double-rotate about Dub.



Final Example:
Delete Naomi

It's a Zig-Zag, so double-rotate about Dub.



Final Example:
Delete Naomi

Done!