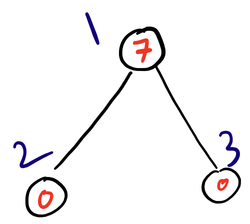
# Flow Finder

- Problem Summary: Given a rooted tree with leafs being sources and some values missing is there a unique solution to assigning unknown values

- First: for all nodes we have two values that could be of importance
  $a_i$ is given input
  $S_i$ is the sum of all leaf values in that subtree

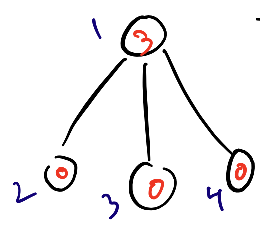
- Observation 1: if a subtree has more than 1 blank node the answer is not unique

e.g.




In this example Node 2 & Node 3 are blank
one possible solution is $a_2 = 2$ & $a_3 = 5$ } not unique
or $a_2 = 3$ & $a_3 = 4$

- observation 2: The previous observation is incomplete.




There is 3 blank leafs but our answer is infact unique since $a_2 = a_3 = a_4 = 1$ whenever we have a missing sum of x and we have x blank nodes they are all 1.

- From these two observation we can create a solution by keeping track of $S_i$ and $M_u$ = set of blank leafs in subtree u

For all Nodes u we have some cases

**case 1: u is a leaf**

if $a_u = 0$ then $M_u = \{u\}$ $S_u = 0$
else $M_u = \{\}$ $S_u = A_u$

**Case 2. u is No a leaf**

$S_u = \sum_{v \in \text{children}(u)} S_v$ $M_u = \bigcup_{v \in \text{Child}(u)} M_v$

if $A_u = 0$ { we can "ignore" this Node
else {
if $M_u$ has one element set it to to $A_u - S_u$
if $M_u$ has $A_u - S_u$ elements set all to 1
otherwise its not possible
}
after make sure to set $S_u = A_u$

- After setting all the leafs then just reconstruct all $a_i$

- Last small detail how do we merge the $M_i$ sets. Naively we end up with $O(n^2)$

- It isn't too dificult. when merging at all children set $M_u$ = largest $M_v$ where v is a child and merge all other sets to that set. Complexity is $O(n \log n)$