CSE 461 Homework 1

Ken Lin

006198682

Total points: 34 points assumed

3, (4 pt assumed) Consider a chain of processes P1, P2, ..., Pn implementing a multitiered client-server architecture. Process Pi is client of process Pi+1, and Pi will return a reply to Pi-1 only after receiving a reply from Pi+1. What are the main problems with this organization when taking a look at the request-reply performance at process P1?

First, the architecture is prone to massive failure from the failure of a single process, because process Pi will only return a reply after receiving a reply from Pi+1, it means that when Pi fails every process with a number less than i will also fail. For example, if the last process in the chain (the one with the highest 'i') fails, then the entire architecture fails, because no process will be able to send a message to processes below them due to the last process never sending a message to the second to last process.

Second, performance will be highly bottled-necked by the last process. Because the last process is used during every request, performance of the entire system will be tied to the processing power of the last process and the bandwidth of the network leading to the last process.

Third, because a request needs to be passed up the entire chain and back again, there's a large latency due to message passing.

Fourth, assuming failure correlates with the number of times a process is called, the more critical areas of the system are more likely to fail. (Because the last process is called the most it has the highest chance to fail, but the last process failing also causes the biggest problem, because the entire system would fail with it.)

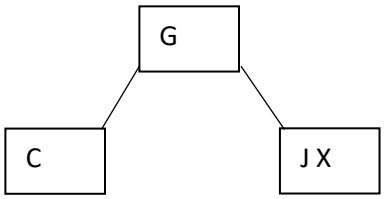
4, (10 pt assumed) Show the B-trees of order four resulted from loading the following sets of keys (each letter is a key) in order:

minimum 2 children per non-leaf, non-root node, will be left-leaning

Will be using https://www.cs.usfca.edu/~galles/visualization/BTree.html to draw the B tree.

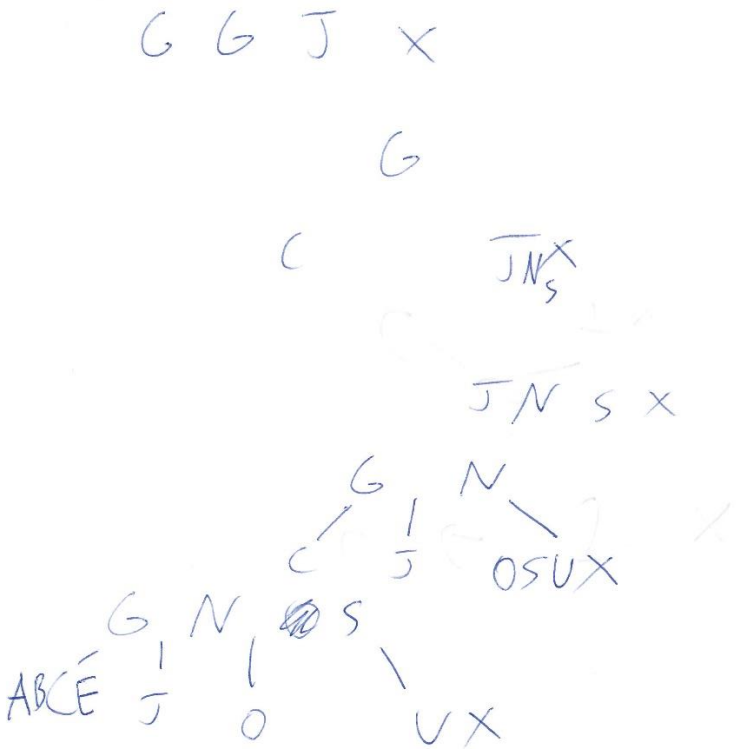
Will be scanning scratch work for part b, rest will only have the final result.

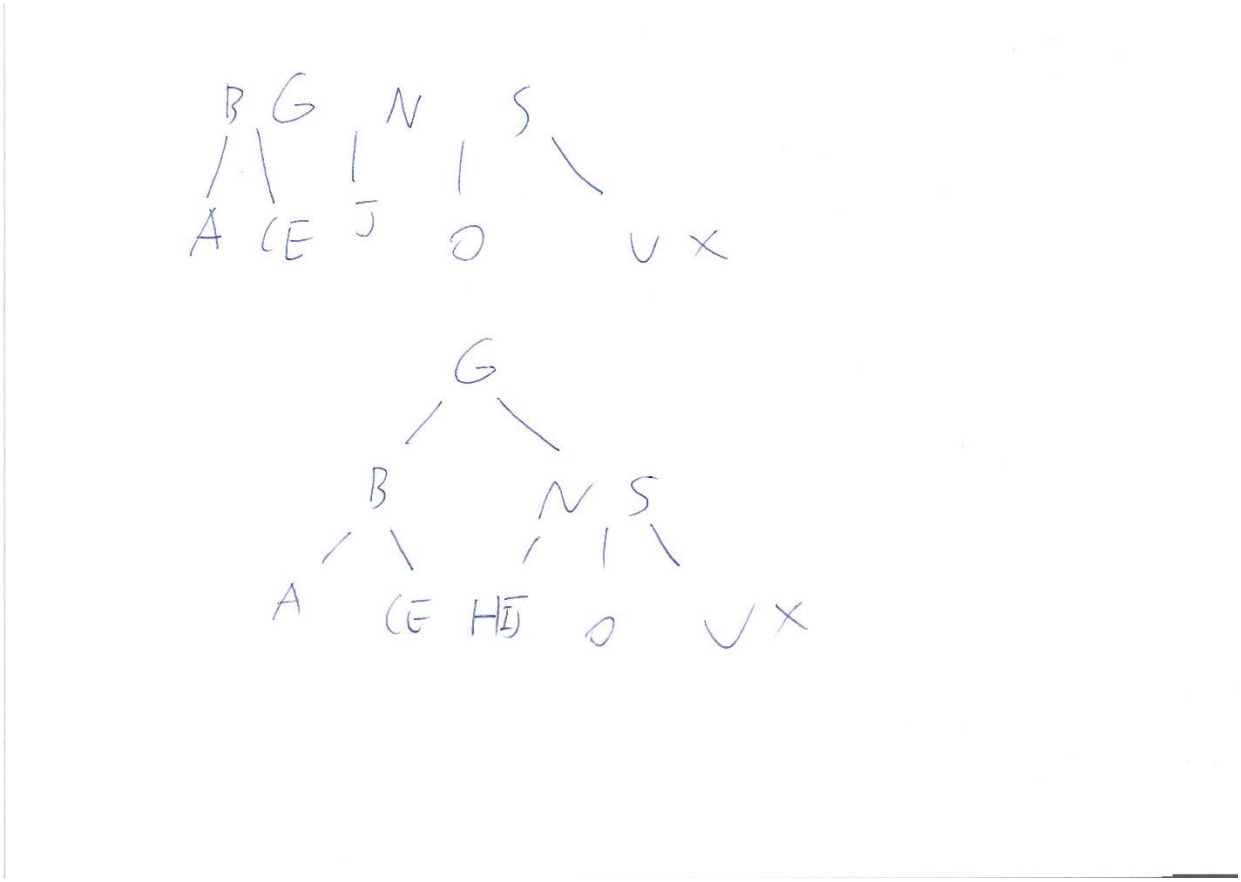
a, C G J X



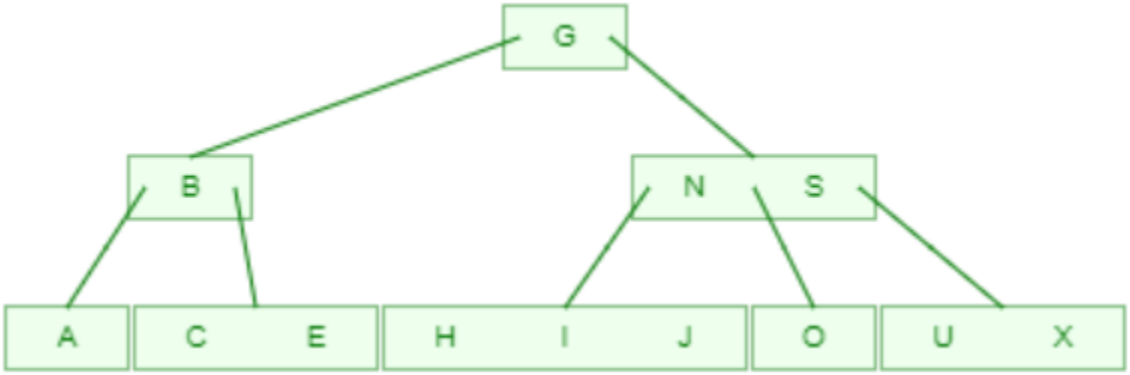
b, C G J X N S U O A E B H I

Scratch work:

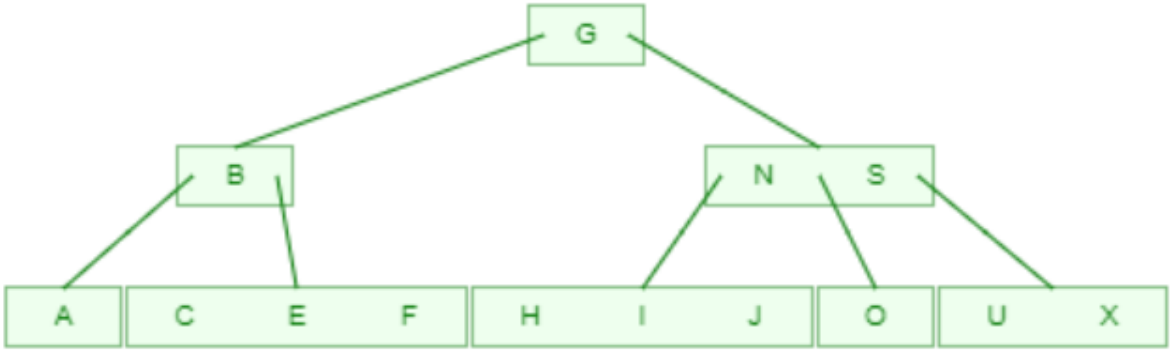




Final tree:

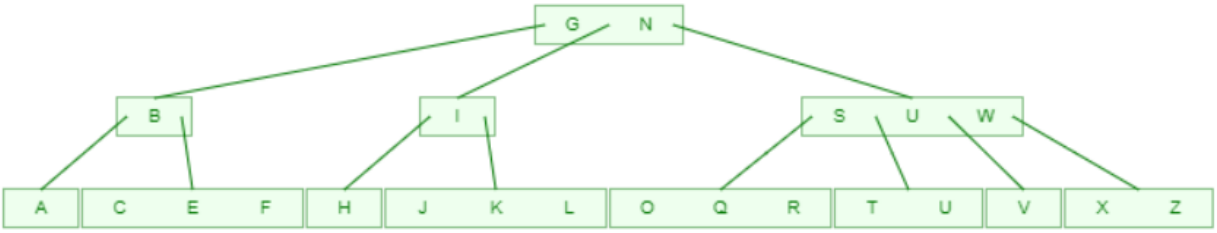


c, C G J X N S U O A E B H I F



d, C G J X N S U O A E B H I F K L Q R T V U W Z

U is repeated. Visualizer places it before the first ‘U.’ (Seems to be consistent with left-leaning.)



5, (10 points assumed) Given a B-tree of order 256,

a, What is the maximum number of children from a node?

256, since that’s the order. (255 is the maximum number of keys.)

b, Excluding the root and the leaves, what is the minimum number of children from a node?

Minimum is 256/2 = 128. That is, non-root, non-leaf nodes have a minimum of 128 children. (And 127 keys.)

c, What is the minimum number of children from the root?

The root is allowed to have no children, so, 0. (The root is also allowed to have no entries, for a b-tree containing no data.)

d, What is the maximum depth of the tree if it contains 100 000 keys?

The root has a minimum of 1 key if the tree is non-empty. We therefore calculate the depth by assuming each node has the minimum number of children for 99,999 keys.

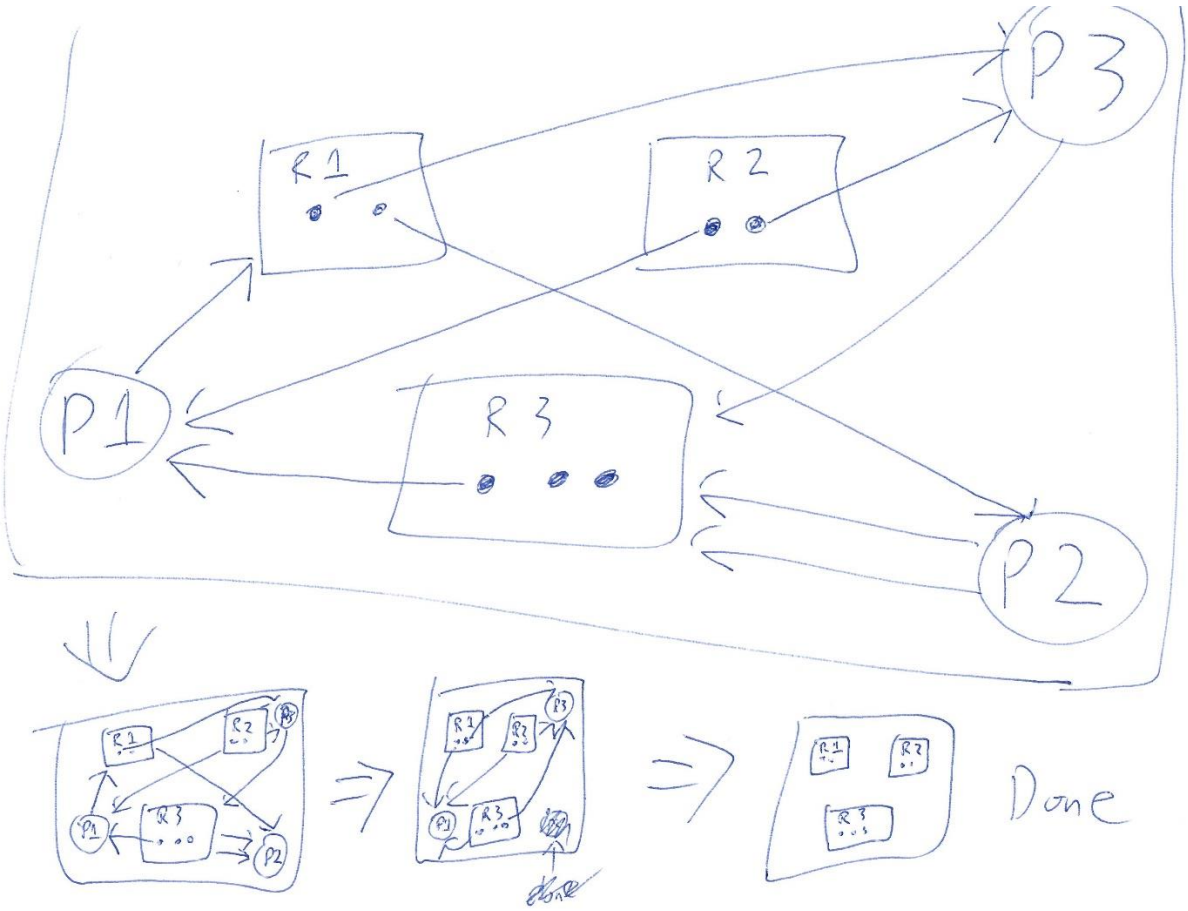
128 is the minimum number of children, so log(99,999)/log(128) = 2.37 gives the depth aside from the root. Rounding up, that's 3 depth for the non-root nodes.

The maximum depth of the tree is 3.

6, (10 points assumed) Construct a general resource graph for the following scenario and determine if the graph is completely reducible: R1, R2, and R3 are reusable resources with a total of two, two, and three units. Process P1 is allocated one unit each of R2 and R3 and is requesting one unit of R1. Process P2 is allocated one unit of R1 and is requesting two units of R3. Process P3 is allocated one unit each of R1 and R2 and is requesting one unit of R3.

This is actually something from next lecture, but it shouldn't be a problem as long as I read ahead.

Steps:



First, P2 is allocated 2 units of R3, which allows it to complete. After P2 terminates, P3 is allocated 1 unit of R3 and P1 is allocated 1 unit of R1. Both P1 and P3 now has enough resources to complete.