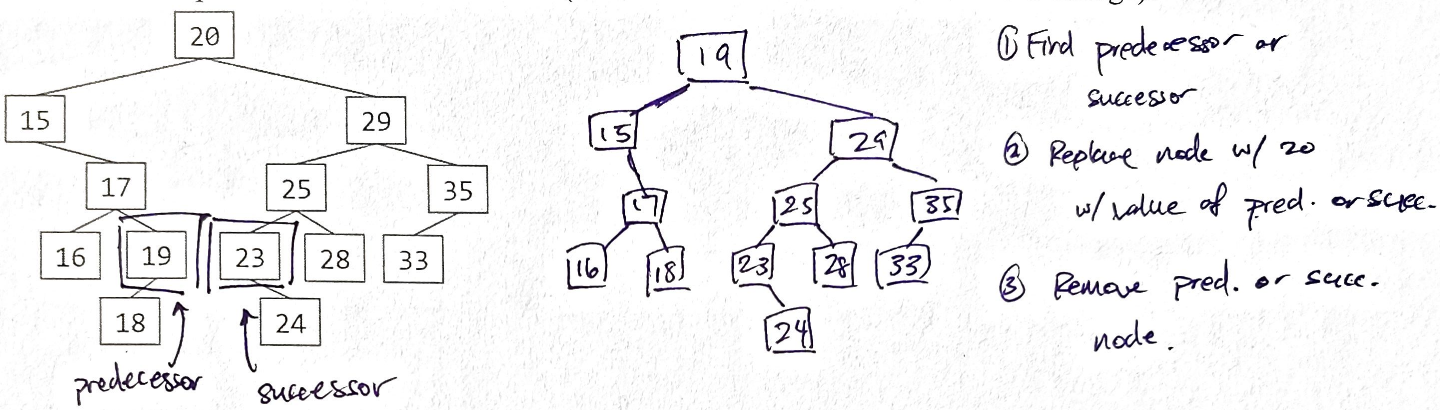
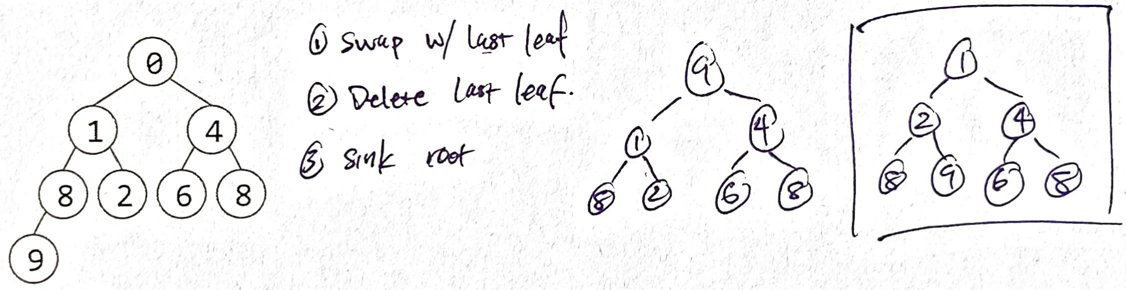
0. Another half point. (0.5 points). Write your name, login, and ID on the front page. Write your exam room. Write the IDs of your neighbors. Write the given statement and sign. Write your login in the corner of every page. Enjoy your free half point. ☺

1. **Basic Operations (6 Points).**

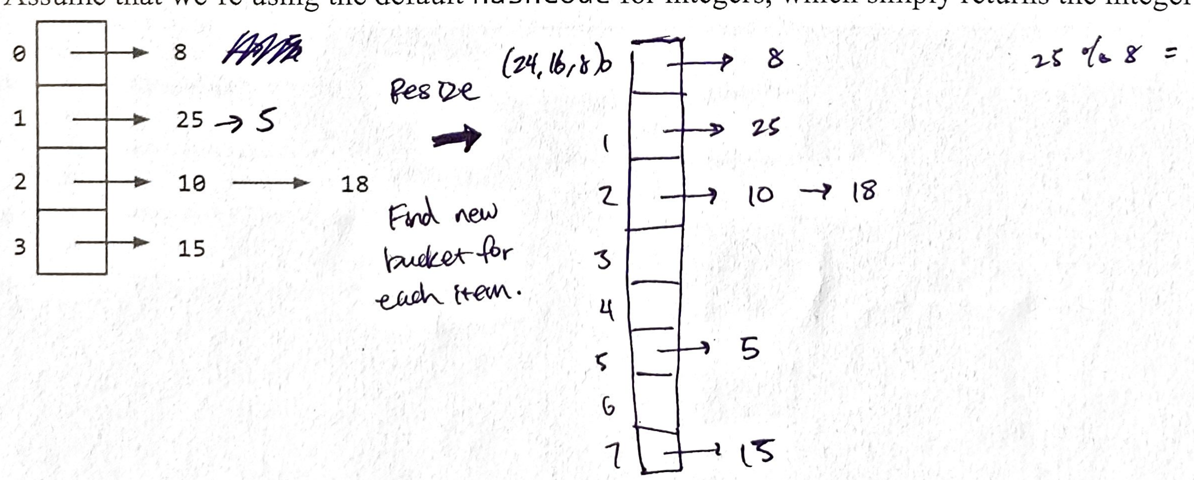
a. **To the right of the BST below**, draw a BST that results if we delete 20 from the BST. You should use the deletion procedure discussed in class (i.e. no more than 4 references should change).



b. **To the right of the minHeap below**, draw the minHeap that results if we delete the smallest item from the minHeap.

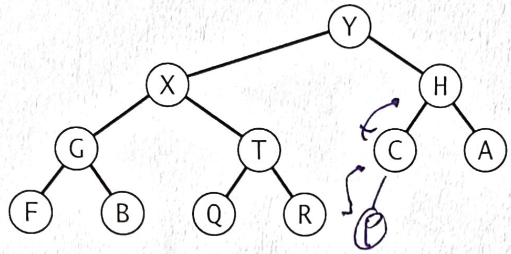


c. **To the right of the External Chaining Hash Set below**, draw the External Chaining Hash Set that results if we insert 5. As part of this insertion, you should also resize from 4 buckets to 8 (in other words, the implementer of this data structure seems to be resizing when the load factor reaches 1.5). Assume that we're using the default hashCode for integers, which simply returns the integer itself.



5. **Binary heaps. (6 points)**

(a) Consider the following binary tree representation of a max-heap.



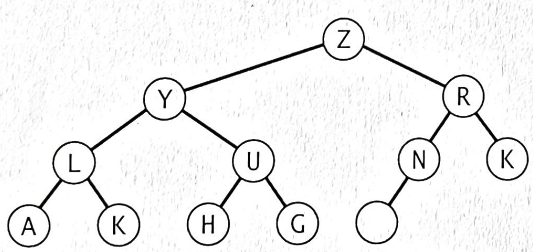
Give the array representation of the heap.

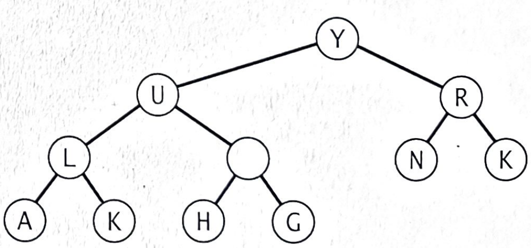
0	1	2	3	4	5	6	7	8	9	10	11	12
–	Y	X	H	G	T	C	A	F	B	Q	R	–

(b) *Insert* the key P into the binary heap above, *circling* any entries that changed.

0	1	2	3	4	5	6	7	8	9	10	11	12
–	Y	X	P	G	T	H	A	F	B	Q	R	C

(c) A *delete-the-max* operation in the binary heap at left results in the binary heap at right.





? ≤ N ? < U ? ≥ H ? ≥ G

Which of the keys below could be the one labeled with a question mark?
Circle all possibilities.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Name:______________________________

3. (11 points) Consider a hashtable with separate chaining with N buckets and k items currently in the table.

(a) k/N is the definition of a term used when discussing hashing. What is this term?

Load Factor

(b) Is it necessary that $k < N$?

No, we can have more items than buckets due to separate chaining

(c) What is the average number of items in a bucket?

k/N

(d) In the worst-case, how many items could be in a single bucket?

k

(e) If $k > N$, is it possible that any buckets are empty?

Yes, if there are collisions

(f) If we resize the table to a table of size $2N$, what is the asymptotic running time in terms of k and N to put all the items in the new table?

Θ(k) time to rehash every item and find its new place.

We don't need to check .equals on the items in the external chain since we know the hash table only contains unique items before resizing.

If there are more buckets than items, Θ(N) time

Simple answer: Θ(N+k)

* What is $\log_2 N$? "$\log_2 N$ is the number of times I need to divide N by 2 to get down to 1." → $\log_2 16 = 4$ because $16/2/2/2/2 = 1$

(b) For each function on the left, give the best matching order of growth of the *running time* on the right.

__B__

```
public static int f1(int N) {
    int x = 0;
    for (int i = 0; i < N; i++)
        x++;
    return x;
}
```

__D__ (print Party example)

```
public static int f2(int N) {
    int x = 0;
    for (int i = 0; i < N; i++)
        for (int j = 0; j < i; j++)
            x++;
    return x;
}
```

_____

```
public static int f3(int N) {
    if (N == 0) return 1;
    int x = 0;
    for (int i = 0; i < N; i++)
        x += f3(N-1);
    return x;
}
```

__C__ (merge sort example)

```
public static int f4(int N) {
    if (N == 0) return 0;
    return f4(N/2) + f1(N) + f4(N/2);
}
```

__B__ (Formula sheet)

```
public static int f5(int N) {
    int x = 0;
    for (int i = N; i > 0; i = i/2)
        x += f1(i);
    return x;
}
```

__E__ (f3 example)

```
public static int f6(int N) {
    if (N == 0) return 1;
    return f6(N-1) + f6(N-1);
}
```

__A__ (Formula sheet)

```
public static int f7(int N) {
    if (N == 1) return 0;
    return 1 + f7(N/2);
}
```

A. $\log N$

B. N

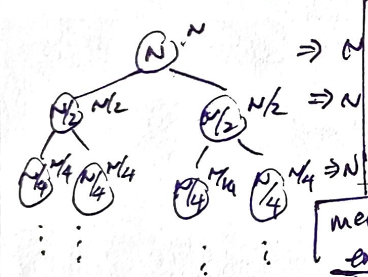
C. $N \log N$

D. N^2

E. 2^N

F. $N!$

$f1(N) \in \Theta(N)$



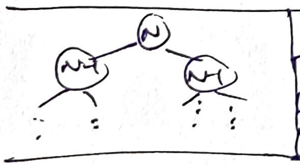
# of levels ≈ $\log_2 N$

work / level ≈ N

$i = N,\ i = N/2,\ i = N/4, \ldots$

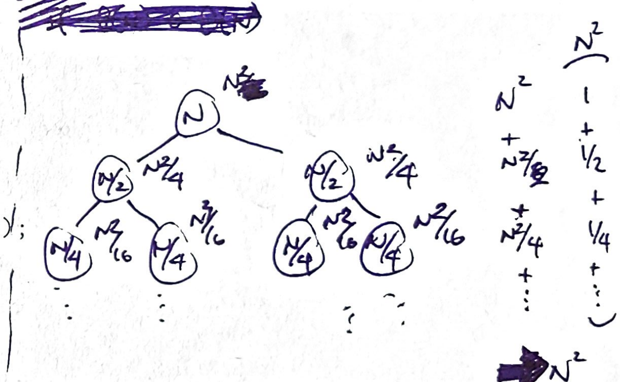
$N + N/2 + N/4 + \cdots \approx N$

$N(1 + 1/2 + 1/4 + \ldots) \approx N$



constant work. $\log_2 N$ levels → $\log_2 N$

OR If $f1(N) \in \Theta(N^2)$

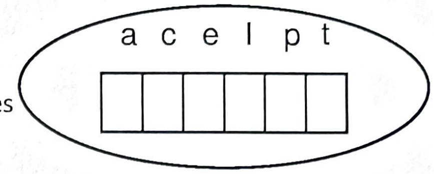


$i = N,\ i = N/2,\ i = N/4,\ i = N/8 \ldots$

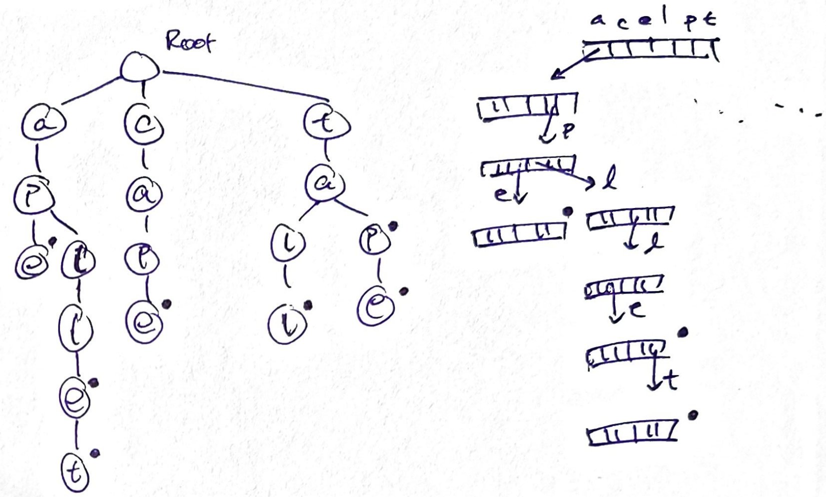
$N^2 + N^2/4 + N^2/16 + \cdots$ → N^2

UCSD CSE 100 Winter 2019

Multi-way trie = Trie with DataIndexedCharMap



1. Create the multiway trie for the following keys (drawing the nodes as shown), inserted in this order (once you get the hang of it, you don't have to draw the *whole* thing):
   apple, ape, applet, cape, tall, tap, tape



2. Does the structure of the trie depend on the order in which you inserted the words?
   A. Yes (B.) No → Each node maintains its own DataIndexed CharMap.
   All level 1 nodes are first chars, all level 2 nodes second chars, ...

3. Assume you insert 100 keys of length 5 digits into a multi-way trie. How tall is your trie in the worst case? 5

4. If N is the number of keys you insert in your trie, and D is the length of the longest key, what is the maximum height of your trie in terms of N and/or D in the worse case?

   D → $\Theta(D)$

5. Assume an alphabet of 6 characters, as above. From this alphabet, you can create 6^5 =7776 different keys (strings) of length 5. If you insert all 7776 of these strings into a MWT and a BST, which will be shorter? How tall is each? Is this true in general (consider different size alphabets, different length strings)? You can assume the BST is perfectly balanced.

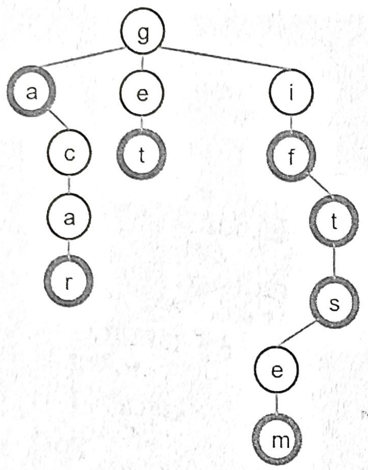
   MWT will be of height 5. BST will be of height $(\log_2 7776) > 5$.

   If the length of keys $> \log_2$ (# of keys), then BST will be better.

6. What is the main drawback of a MWT compared to a BST, especially when the MWT isn't full?

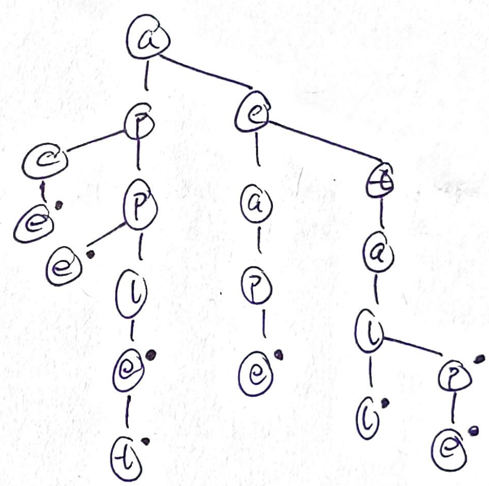
   Lots of unused space, wasted DataIndexed Char Map.

7. List all of the words you can find in this TST. Nodes that end words have extra-thick circles.



a
car
get
if
it
its
item

8. Create the TST for the following strings (same set as on the last page)
apple, ape, applet, cape, tall, tap, tape



9. If you add the word "ace" to the TST you just created, how many nodes do you have to add?

2

10. Does the structure of a TST depend on the order in which the keys were inserted?

Yes.