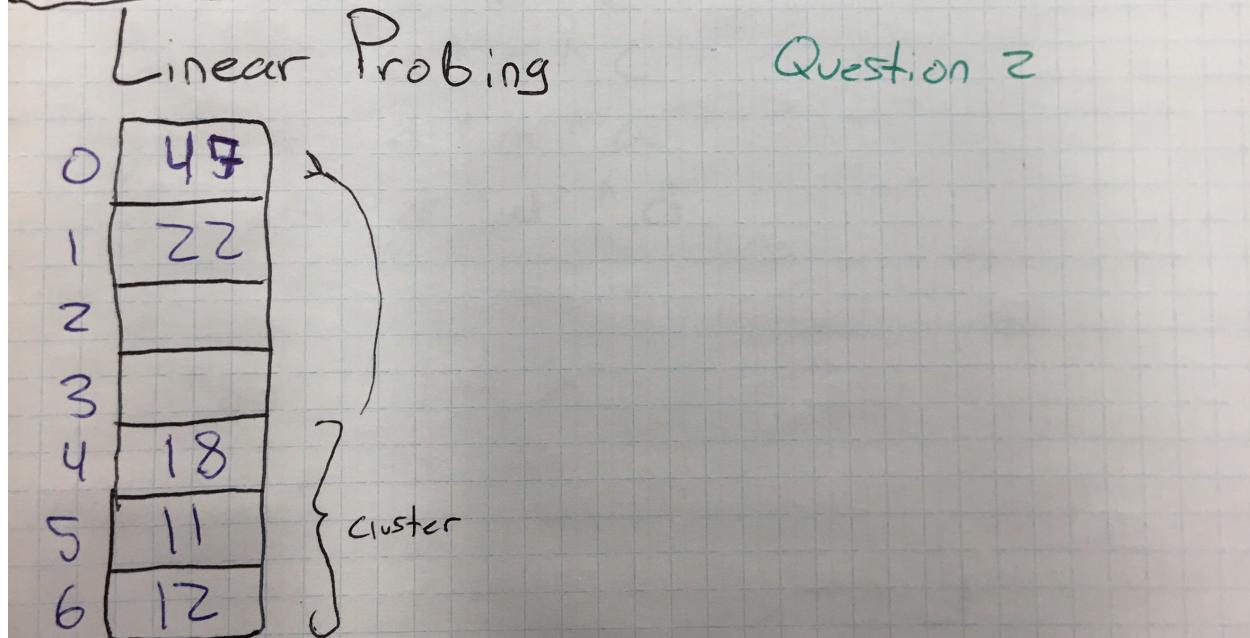
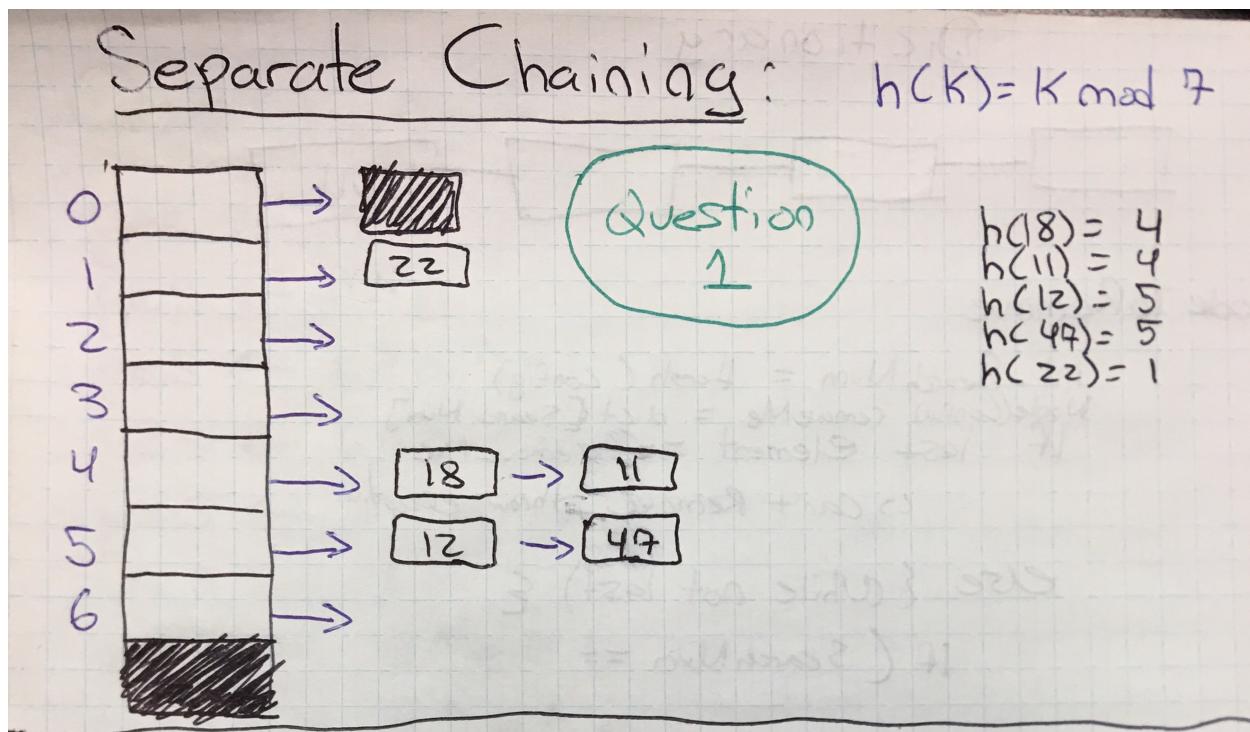


Firas Aboushamalah

250-920-750

Assignment #3

Question 1 & 2



Question 3:

Double Hashing:	
$h(K) = K \bmod 7$	$h'K = 5 - (K \bmod 5)$
$h(18) = 4$	
$h(11) = 4$	$h'K = 5 - (11 \bmod 5)$ $= 5 - 1 = 4$
$h(12) = 5$	
$h(47) = 5$	$h'K = 5 - (47 \bmod 5)$ $= 5 - 2 = 3$
$h(22) = 1$	$h'K = 5 - (22 \bmod 5)$ $= 5 - 2 = 3$
$h(18)$	
$h(12)$	

0 47
1 11
2
3 22
4 18
5 12
6

Question 4:

$$\begin{aligned}f(n) &= f(n-1) + c_2 n + c_3 && \text{Sub } f(n-1) \text{ for } f(n) \\f(n-1) &= (f(n-2) + c_2(n-1) + c_3) + c_3 \\f(n-1) &= f(n-2) + c_2[(n-1)+n] + 2c_3 && \text{Sub } f(n-2) \text{ for } f(n-1) \\f(n-2) &= (f(n-3) + c_2(n-2) + c_3) + c_2(n+n-1) + c_3 \\&\vdots \\&= f(0) = c_2[n + (n-1) + \dots + 1] + nc_3 && f(0) = c_1 \\&= c_1 + c_2[n + (n-1) + \dots + 1] + nc_3 && \text{Base case} \\&= c_1 + c_2\left[\frac{n(n+1)}{2}\right] + nc_3 \\&= c_1 + c_2\left[\frac{n^2+n}{2}\right] + nc_3 \\&= c_1 + \cancel{\frac{c_2 n^2}{2}} + \cancel{\frac{c_2 n}{2}} + nc_3 && \text{Cancel out coefficients}\end{aligned}$$

$$\therefore n^2 + n + n$$

time complexity of $f(n)$ is $O(n^2)$

Question 5i:

```
Algorithm maxValue(r)
In: Root r of tree
Out: Largest int value stored in node of tree.

IF r.isLeaf then return r;
else {
    max ← 0;
    For each Child c of r do
        Preorder(r);
        IF c.value > max then {
            max ← c.value
        }
    }
    return max;
```

Question 5ii:

```
C1 IF r.isLeaf then return r;
else {
    C2 max ← 0;
    C3 × n } For each Child c of r do
        Preorder(r);
        IF c.value > max then {
            max ← c.value
        }
    }
    C4 return max;
```

$$f(0) = C_1$$

$$f(n) = C_2 + C_3 n + C_4$$

Add up the constants for size $n = 0$ and the algorithm ends at first line so $f(0) = C_1$

For size n , the algorithm has 2 constants and a 3 constant multiplied by n iterations via "preorder(r)"
Therefore, the time complexity of this algorithm is $O(n)$ for $n > 0$