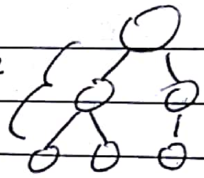
Ashlyn Goh 1002840 CI02

Ashlyn Goh 1002840

50.004 Problem Set 1 - Date No.

Exercise 1 Q1 Minimum number of elements = 2^h

Maximum number of elements = $2^{h+1}-1$

2 

Q2 Increasing order (heap sort) ~~for~~ $\Rightarrow \theta(n\lg n)$

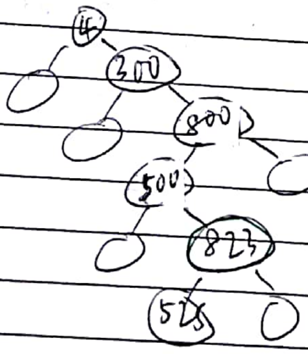
~~E~~ Even though array is already sorted, ~~each~~ all elements have to be inserted into a heap and extracted again ~~to~~ ~~a~~ for heap sort.

Decreasing order $\Rightarrow \theta(n\lg n)$

~~Every~~ ~~All~~ A heap will be built and ~~elements~~ ~~elements~~ maximum element will be removed ~~each time~~ with max-heapify called ~~the~~ each time.

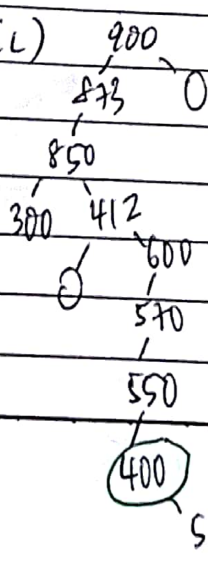
Q3 In a min heap, the biggest element would reside in one of the leaves nodes. This is because the children of a node are bigger than the node itself, hence one of the leaves contain the largest element in a min heap.

Exercise 2 (A)



Since node 500 lies to the left of node 800, the children of node 500 must ~~be~~ all be less than 800. Hence, it violates the property of Binary Search Trees when node 823 is a child of node 500. Therefore, option A is impossible.

(C)



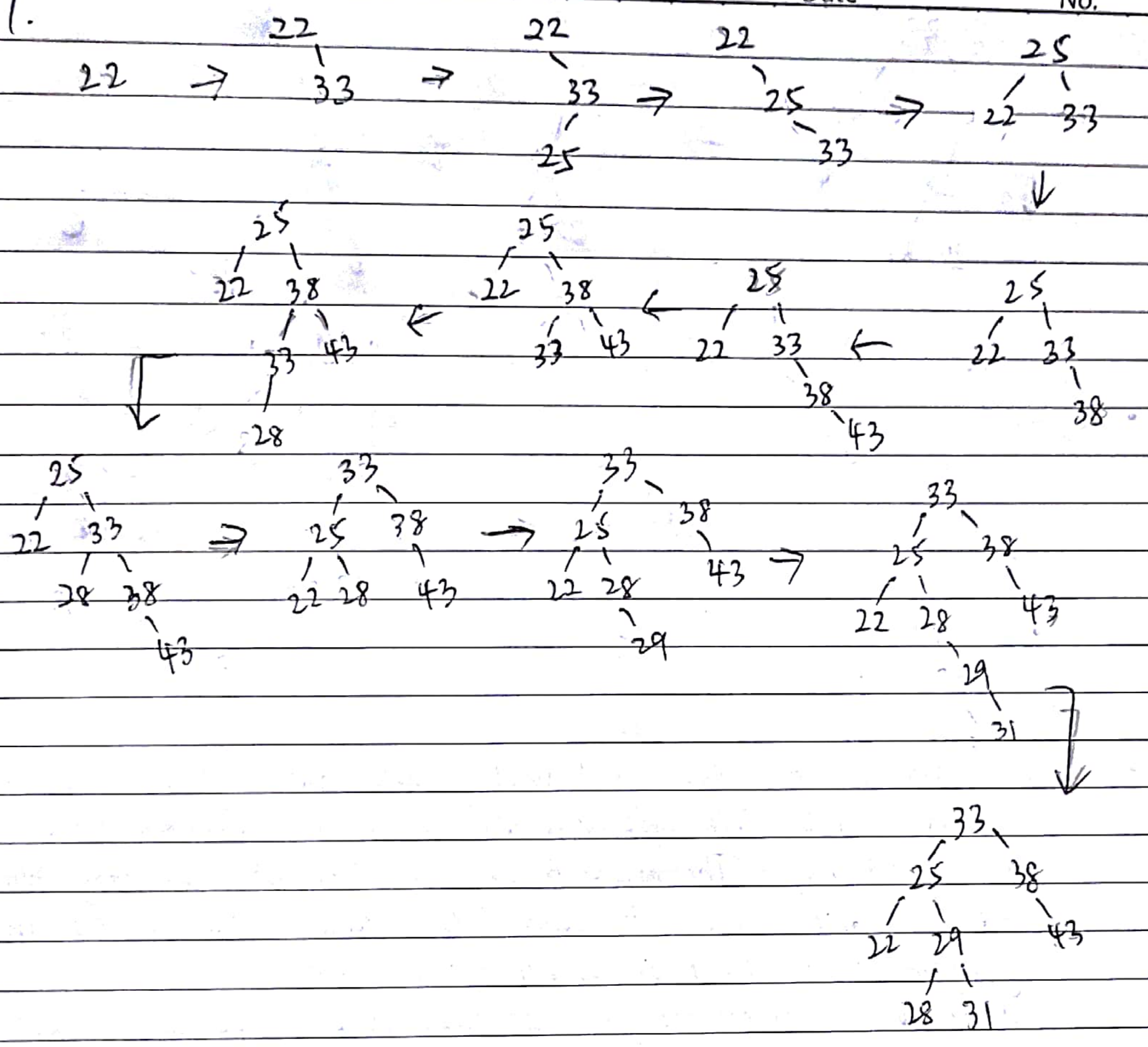
Since node 600 lies to the right of node 412, the children of node 600 must all be greater than 412. Hence, it violates the property of Binary Search Trees when node 400 is a child of node 600. Therefore, option C is impossible.

(A) and (C) //

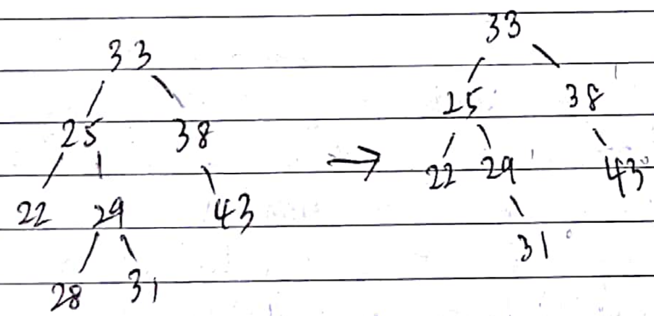
Date No.

Exercise 3

1.

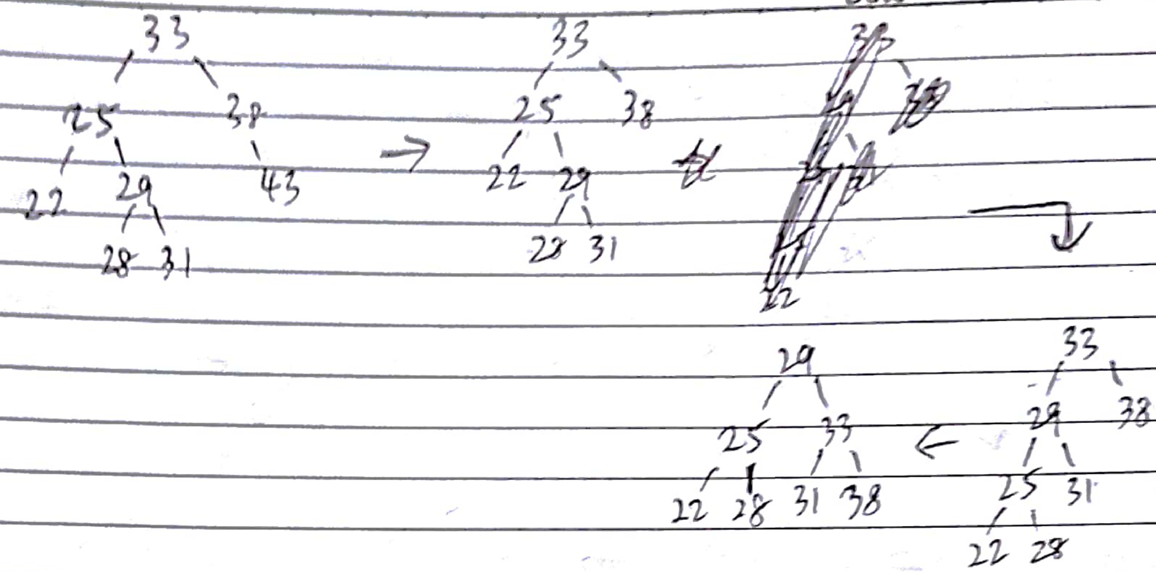


Q2



Date No.

Q3



Ex 4

Q1 Yes, the algorithm still works after the modification. The modified algorithm still places ~~elements~~ each element A[j] into its sorted position in output array B. The array B will still be sorted correctly. However, this algorithm is not stable. An element taken from A later started out with a higher index than one taken earlier. Hence, numbers with the same value do not appear in the output array B in the same order as they do in input array A.

(taken from CLRS book)

Q2 We can preprocess the input using counting sort, such that ~~intermediate array,~~ C[i] gives the number of elements less than or equal to i in the input array (Array C is the intermediate array in counting sort). To find out how many integers fall into a range [a...b], we can then use C[b] − C[a−1]. (If a is 0, just C[b]) This takes O(1) time to compute. Since the preprocessing uses counting sort's algorithm, it takes Θ(n+k).

Date No.

Q3

hat		tea		rag		bat
ten		one		pan		box
hen		rag		hat		hat
two		ten		rat		hen
pan		hen		bat		one
one	→	pan	→	tea	→	pan
tea		two		ten		rag
rat		hat		hen		rat
rag		rat		one		tea
box		bat		box		ten
bat		box		two		two //

~~EX5~~

~~$h'(k,i) = (h(k) + c_1 i + c_2 i^2) \bmod m$~~

~~From the search ~~sheme~~ given, we can see that the probing procedure is as follows : $h(k) \to h(k)+1 \to h(k)+1+2 \to \dots \to h(k)+1+2+3+\dots+$~~

~~To find out if this scheme follows the quadratic probing~~

To find out if this scheme follows the formula for the scheme, we need to find out the

Date No.

Ex5 From the ~~scheme~~ search scheme given, we can see that the probing ~~~~ sequence is as follows:

$h(k) \bmod m \rightarrow (h(k)+1) \bmod m \rightarrow (h(k)+1+2) \bmod m$

$\downarrow$

$(h(k)+1+2+3+\dots+i) \bmod m \leftarrow \dots \leftarrow (h(k)+1+2+3) \bmod m$

To find out if this scheme follows the quadratic probing scheme, we need to determine the formula for the addition portion of the procedure and see if it follows ~~a~~ $c_1 i + c_2 i^2$ formula.

$i = 0 \quad 0 \rightarrow 0$

$i = 1: \quad 1 \rightarrow 1$

$i=2 \quad 1+2 \rightarrow 3$

$i=3 \quad 1+2+3 \rightarrow 6$

$i=4 \quad 1+2+3+4 \rightarrow 10$

Let $ai^2 + bi + c$ be the formula required

when $i=1$, $a+b=1 \Rightarrow a = 1-b$ — ①

when $i=2$, $4a+2b=3$ } sub ① in.

$4-4b+2b=3$

$1 = 2b \Rightarrow b = 1/2$

$a = 1 - \frac{1}{2} = \frac{1}{2}$.

$\therefore$ formula $\Rightarrow \frac{1}{2}i^2 + \frac{1}{2}i$ //

To check

$\frac{1}{2}i^2 + \frac{1}{2}i$.

when $i=3$, $\frac{1}{2}(3)^2 + \frac{1}{2}(3) = 6$

when $i=4$, $\frac{1}{2}(4)^2 + \frac{1}{2}(4) = 10$

(verified that the formula works for other values of i)

Thus, this scheme is an instance of the general quadratic probing scheme.

$h'(k,i) = (h(k) + \frac{1}{2}i + \frac{1}{2}i^2) \bmod m$ //