COP 5536

Advanced Data Structures
Spring 2021

Exam 1
CLOSED BOOK
9:00 am – 10:30 am
(60 Minutes + Extra 30 Minutes for scanning & submission)

PLEASE READ THE FOLLOWING INSTRUCTIONS CAREFULLY

1. For all problems, use only the algorithms discussed in class/text.
2. Write your answers directly on your own white blank paper. You may use extra scratch paper for calculation, but these will not be submitted.
3. All answers will be graded on correctness, efficiency, clarity, elegance and other normal criteria that determine quality.
4. You may use only a pen or a pencil. No calculators allowed.

Note. All answers will be graded on correctness, efficiency, clarity, elegance and other normal criteria that determine quality. The points assigned to each question are provided in parentheses.

Last Name: _____

First Name: _____

UFID: _____

Q. 1 (10)	Q. 2 (12)	Q. 3 (14)	Q. 4 (14)	Total (50)

Question 1 (10):

Solution: Amortized cost of Insert is 2. Amortized cost of Pop is 0.
(2pts for each result, 6 pts for the argument)
Accouting method:
We will maintain the invariant that every item has one credit. Start with an initially empty data ordered stack, we assign two credits to each OrderedPush operation, which covers one for the actual cost and one to satisfy the invariant. Note that the actual cost of removing each item both operations is covered by the remaining credit on the item. Therefore we don't need to assign any extra credits for the Pop operation.

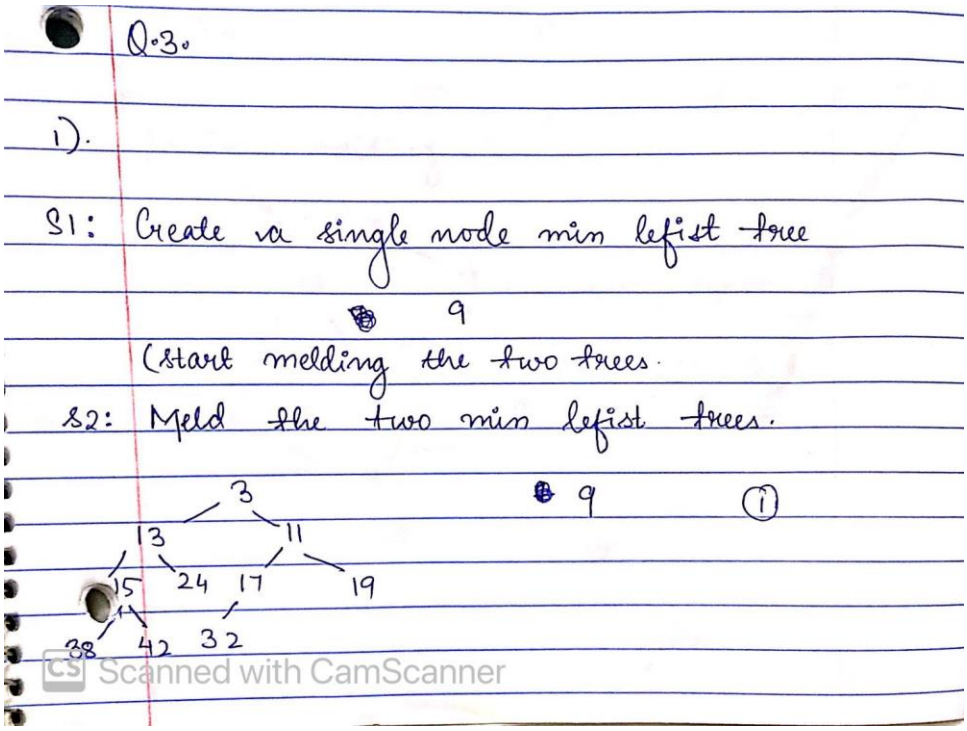
Question 2 (12):

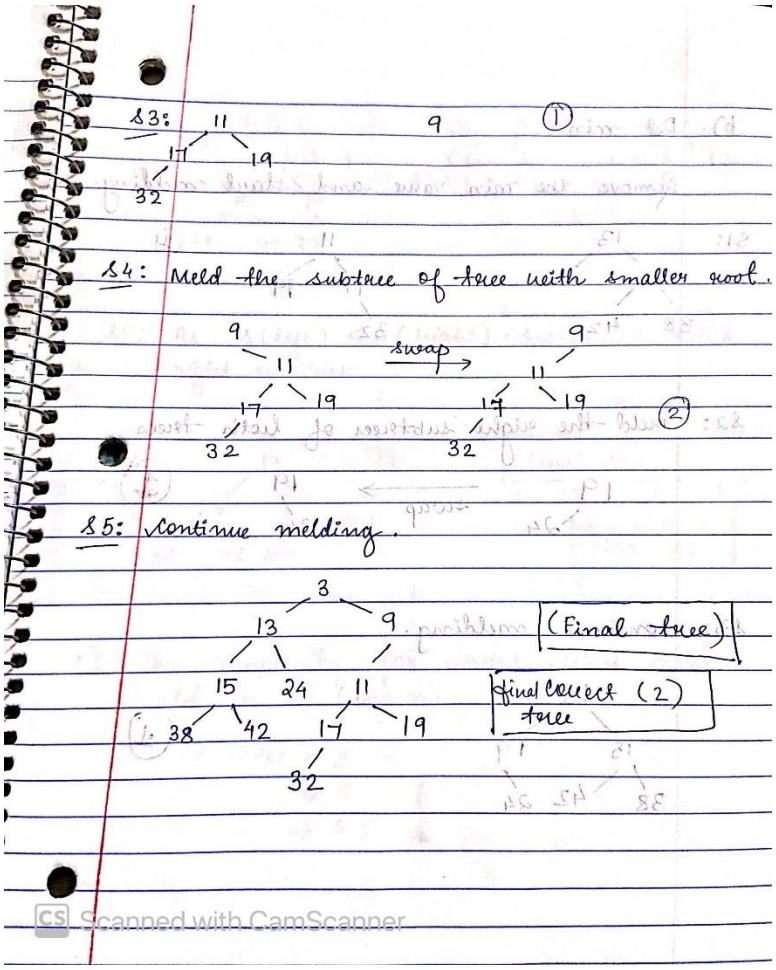
Solution:
Note that block size is typically fixed by the OS. The buffer size b has to match the block size B, i.e., b = B = S/(2k+2).
S = 180, Ts = 6ms, Tl = 4 ms, Tt= 0.1 ms/second
(4pts) B = S / 2k + 2 = 180 / (2 x 8 +2) = 180/18 = 10
Time to read a buffer = (6 + 4 + 10 x 0.1) = 11ms
Number of buffers/pass = 1000/10 = 100
Input time/pass = 11 x 100ms = 1100ms
Number of passes = $log_k^m = \log_8^m$
(8pts, 2pts for each intermediate step above) Total input time = $1100 * \log_8^m$

Question 3 (14):

Solution :

Marking scheme for 3(a), 3(b) will be followed as mentioned in the images below.

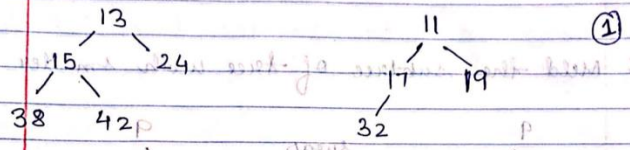




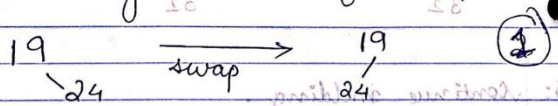
b). Del min:

Remove the min value and start melding.

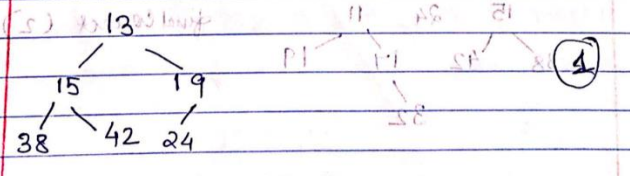
S1:

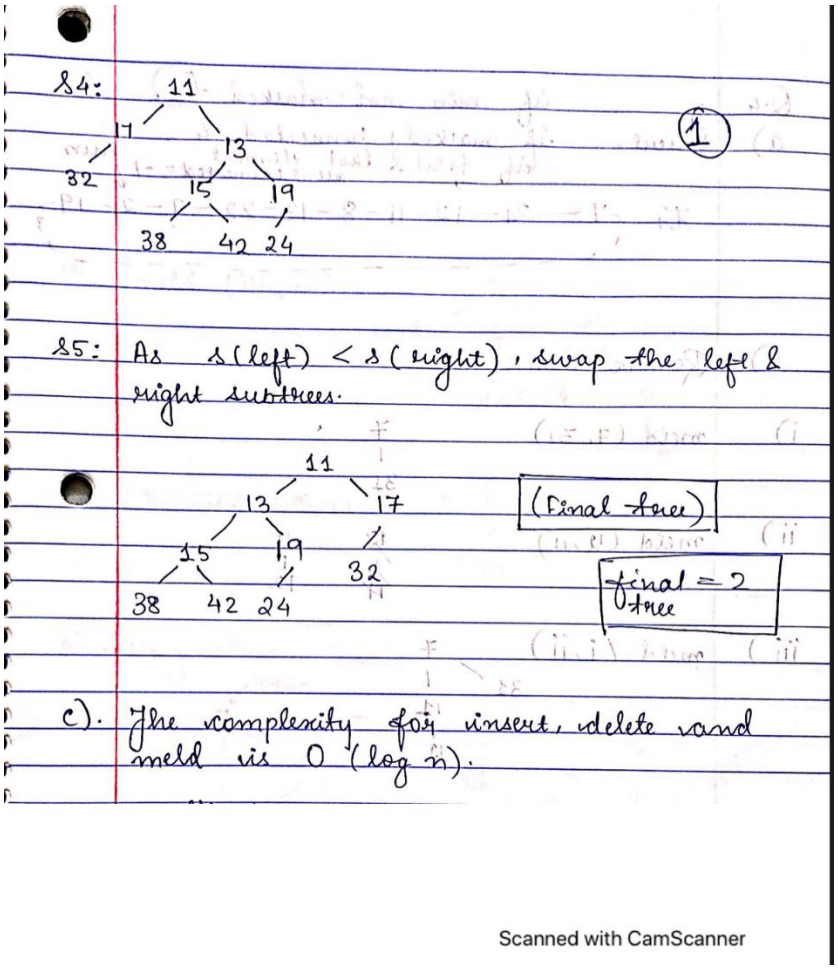


S2: Meld the right subtrees of both trees.



S3: continue melding.





Question 4 (14):

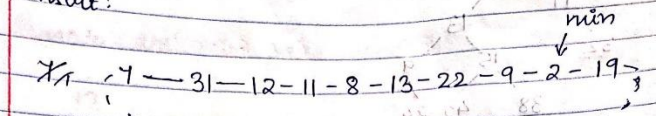
4(a) - If min is not marked : -2
If first and last element are not connected: -1

4(b) - If min is not marked: -1
(Answers may vary as per the combining)

4(c) - If min is not marked: -1

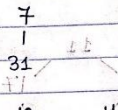
Q.4

a). Insert:

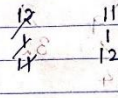


b). Remove min:

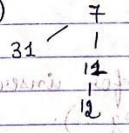
i) meld (7, 31)



ii) meld (12, 11)

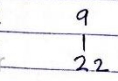


iii) meld (i, ii)

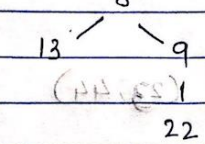


iv) meld (8, 13)

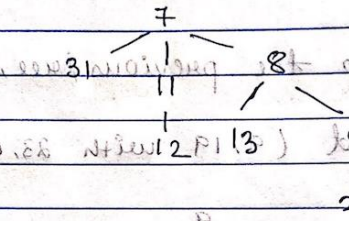
v). meld (22, 9)



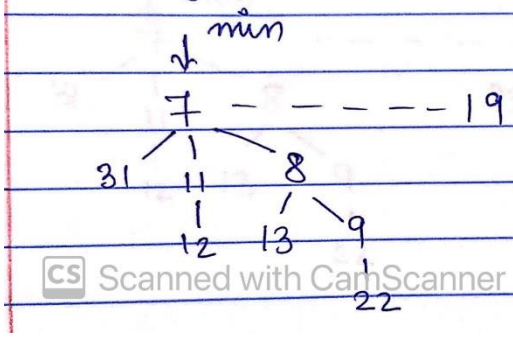
vi) meld (iv, v)



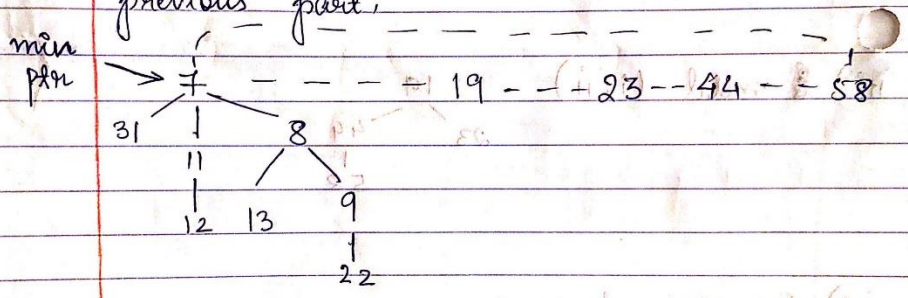
vii) meld (iii, vi)



Resulting tree:



c) Inserting into the tree from the previous part.



NOTE: (Melding will be considered incorrect as it does not result in O(1) complexity.)