

Question 1.

- a. Aggregate method. There are $\lfloor \sqrt{n} \rfloor$ operations that each cost $4\sqrt{k}$, and the other $n - \lfloor \sqrt{n} \rfloor$ operations each cost 1. This yields a total cost of $2 * (1 + 2 + 3 + \dots + \lfloor \sqrt{n} \rfloor) + (n - \lfloor \sqrt{n} \rfloor) * 1 = 2 * ((\lfloor \sqrt{n} \rfloor) * (\lfloor \sqrt{n} \rfloor + 1) / 2) + (n - \lfloor \sqrt{n} \rfloor) = n + \lfloor \sqrt{n} \rfloor + n - \lfloor \sqrt{n} \rfloor = 2n$. So the amortized cost is $2n / n = 2$.
- b. Accounting method. Charge each operation \$1. When k is not a perfect square, use \$1 to pay for the operation and use the extra \$1 for credit. When k is a perfect square, the preceding $(k - 1) - (\sqrt{k} - 1)^2 = 2\sqrt{k} - 2$ operations have each paid a credit of \$1, which together with the \$2 for the current operation yields exactly enough to pay for its $2\sqrt{k}$ actual cost.
- c. Potential method. Define Φ = the number of operations since the most recent perfect square. That is, let $\Phi_k = k - (\lfloor \sqrt{k} \rfloor)^2$. When k is not a perfect square, the amortized cost is $c' = c + \Phi_k - \Phi_{k-1} = 1 + (k - (\lfloor \sqrt{k} \rfloor)^2) - ((k - 1) - (\lfloor \sqrt{k} \rfloor)^2) = 2$. When k is a perfect square, the amortized cost is $c' = c + \Phi_k - \Phi_{k-1} = 2\sqrt{k} + (k - (\sqrt{k})^2) - ((k - 1) - (\sqrt{k} - 1)^2) = 2\sqrt{k} + 0 - (2\sqrt{k} - 2) = 2$.

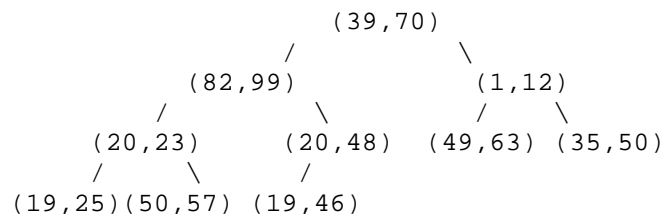
Question 2.

- (a) $m=5, n=4$
 # of dummy nodes needed
 $(1-m) \bmod (n-1) = (1-5) \bmod (4-1) = -4 \bmod 3 = (6-4) \bmod 3 = 2$
 Pass 1: 4 runs with 0, 0, 500, 700 records ==> 1200-record run
 Pass 2: 4 runs with 1200, 900, 1100, 1500 records ==> 4700-record run
- (b) total time = I/O time + Merge time
 Pass 1: $2 * 1200 * 10 / 100 + 12 = 252$ seconds
 Pass 2: $2 * 4700 * 10 / 100 + 47 = 987$ seconds
 Thus, total 1239 seconds

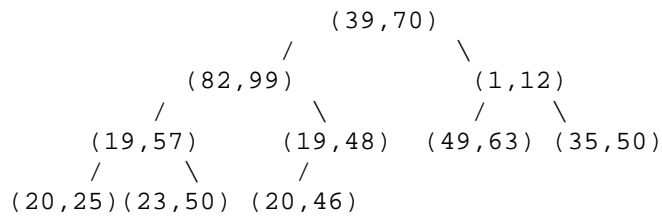
Question 3.

(a)

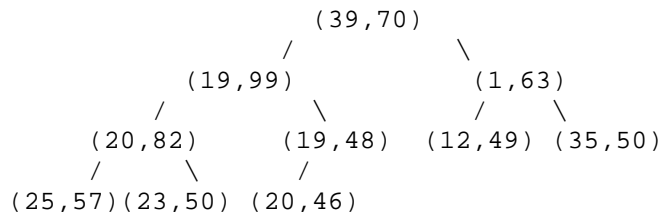
Step 1. Adjust end points in each node if needed



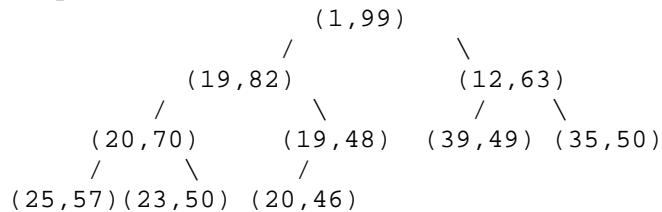
Step 2-1. Starting from the parent of the last node, from right to left and from bottom to top.



Step 2-2.

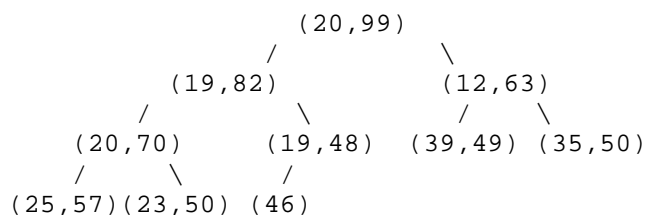


Step 2-3 and Final.

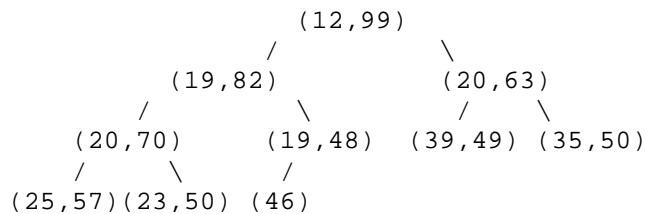


(b)

- i) remove the min element from the root
- ii) remove the left pointer (20) from the last node and reinsert it into the root

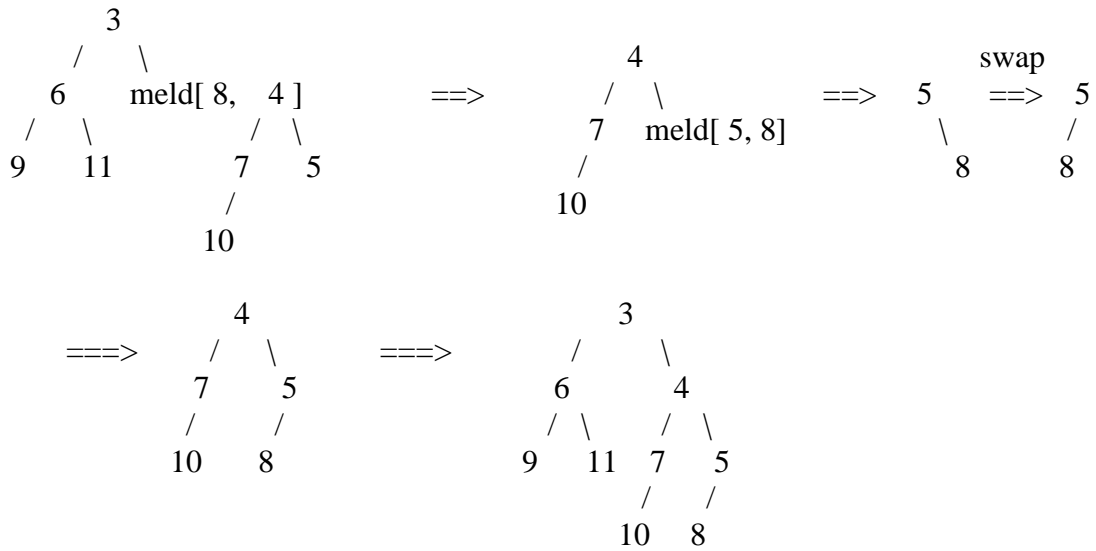


=>



Question 4.

“Meld right subtree of tree with smaller root and all of other tree.”

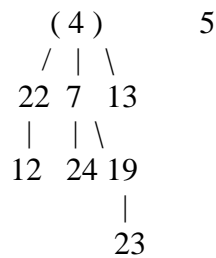


Question 5.

a) Before DeleteMin: () points to the min element

7 12 4 22 (3) 24 13 23 19 5

b) After Deletemin:



c)

$$|B_k| = |B_0| + |B_1| + \dots + |B_{k-1}|$$

$$= 1 + 2 + \dots + 2^{k-1}$$

$$= 2^k$$

$$|B_k| = |B_{k-1}| + |B_{k-1}|$$

$$= 2^{k-1} + 2^{k-1}$$

$$= 2^k$$