19

1. (20) Given a set of n records with distinct keys to be implemented using the given data structures as shown. Fill in the following table with the best possible worst-case complexity $T_w(n)$ in big-O for the corresponding operations. If more than one algorithm exists for a given operation, you must give the complexity of the best possible algorithm.

	build	insert	find	findMax	deleteMin
maxMin Heap	O(n)	O(lgn)	O(n)	O(1)	O(lgn)
2-3 Tree	O(nlgn)	O(lgn)	O(lgn)	O(lgn)	O(lgn)
Min Pairing Heap	O(n)	O(1)	O(n)	O(n)	O(lgn) O(n)
Min Leftist Heap	O(nlgn)	O(lgn)	O(n)	O(n)	O(lgn)

2. Given a min 3-heap H implemented using an array A[0..maxSize] with root at A[0]. Answer the following questions with integer solution.
(a) (3) For a given node x stored at A[2038], x ≠ root, where can we find the parent of x?

$\left\lfloor \frac{2038-1}{3} \right\rfloor = 679$

$3\overline{)2037}$ = 679

5/3 = 1 6/3 = 2 7-1/3 = 2

(b) (3) For a given node x stored at A[2038], where can we find the second child of x?

2038 × 3 + 2 = 6116

(c) (4) If the min 3-heap H has 5138 elements, where can we find the non-leaf node x with the largest array index i? *Remark:* The node x is the first non-leaf node encountered in the reversed level order of H.

0

min

$\left\lfloor \frac{5138}{2} \right\rfloor = 2569$ leaf node number

$\left\lceil \frac{5138}{3} - 1 \right\rceil = 1712$

5138 − 2569 = 2569 non-leaf

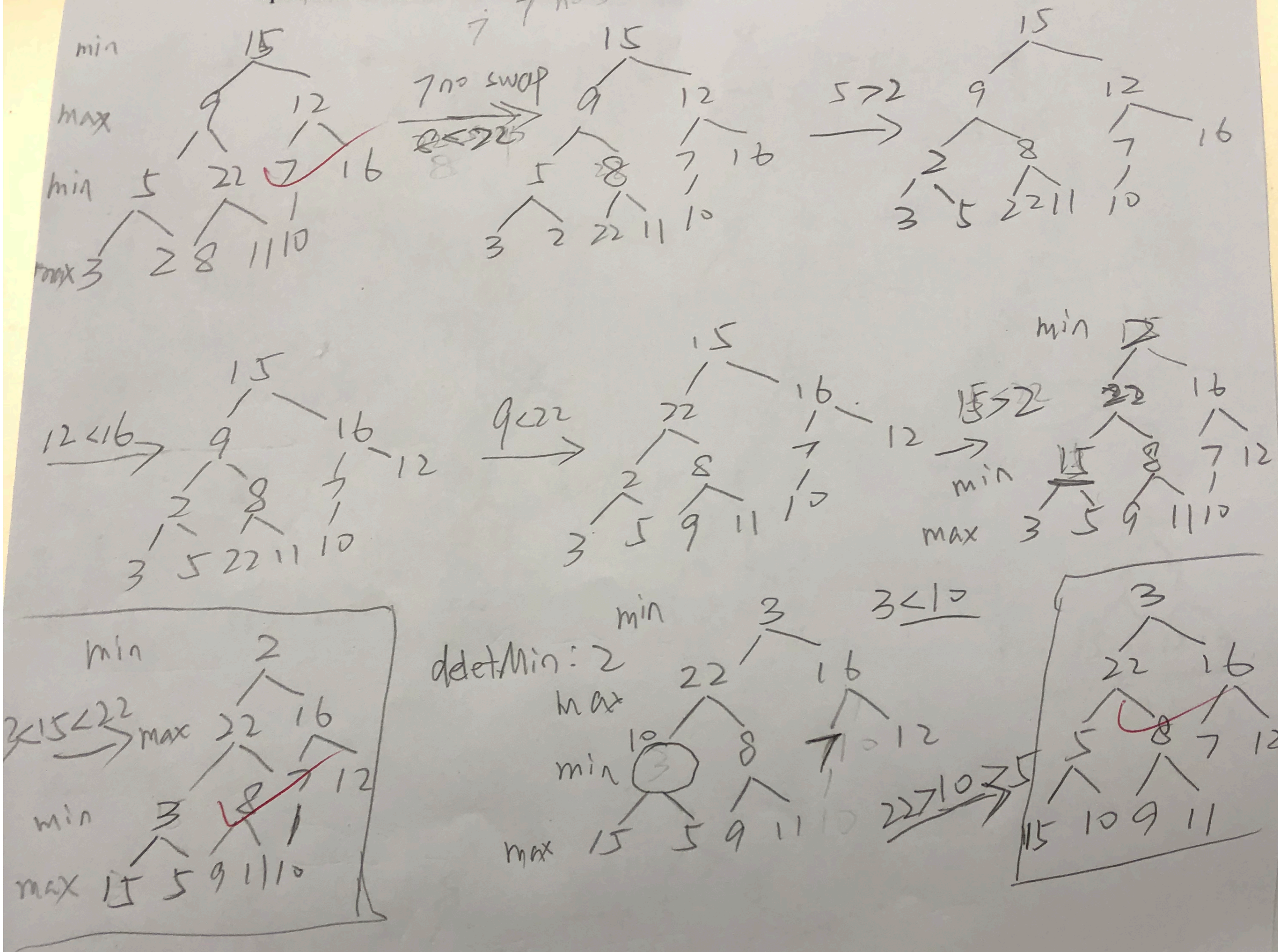
$\frac{5138-2}{k} = 1712$

largest index = 25~~68~~

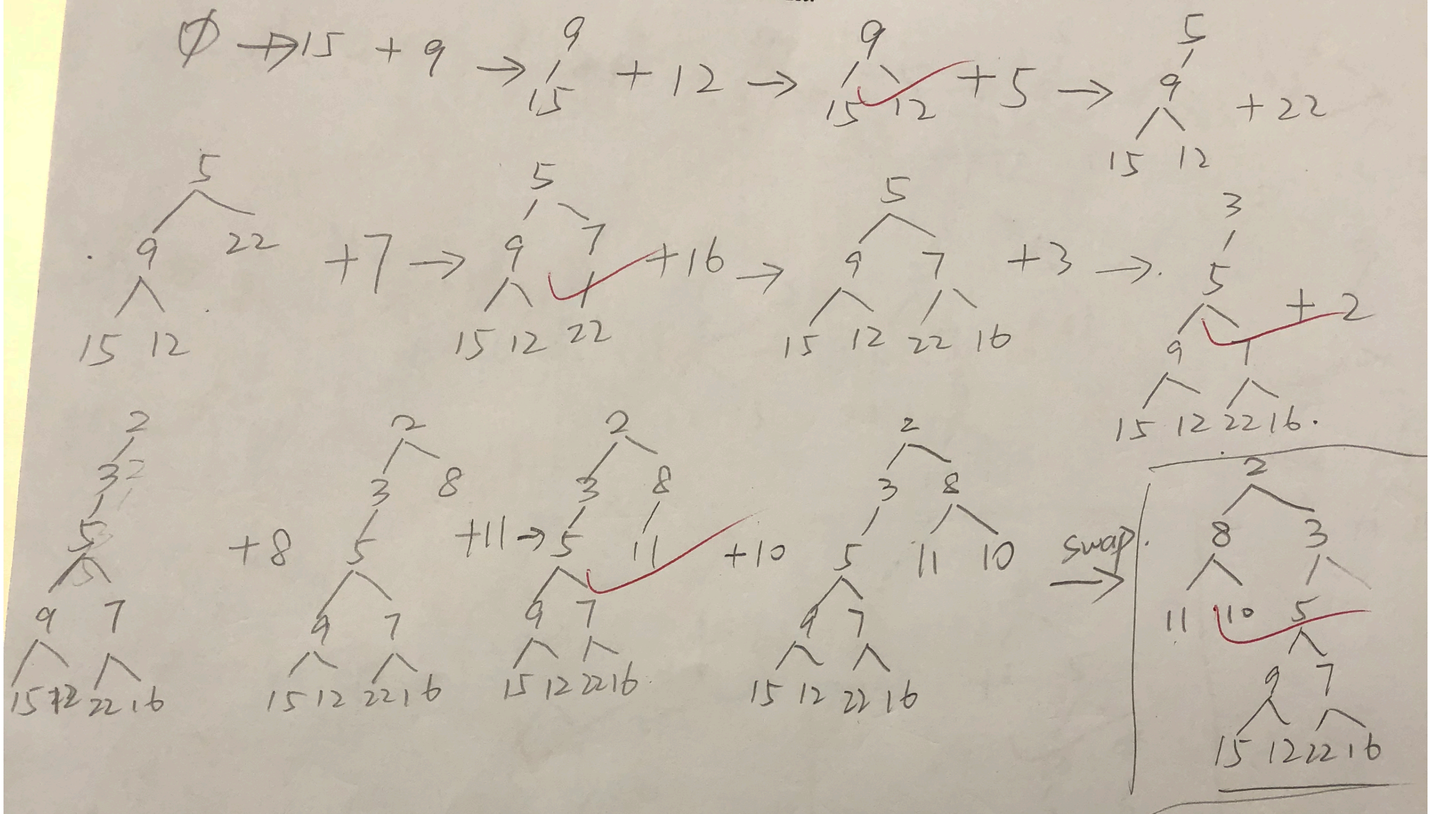
$3\overline{)5136}$ = 1712

12

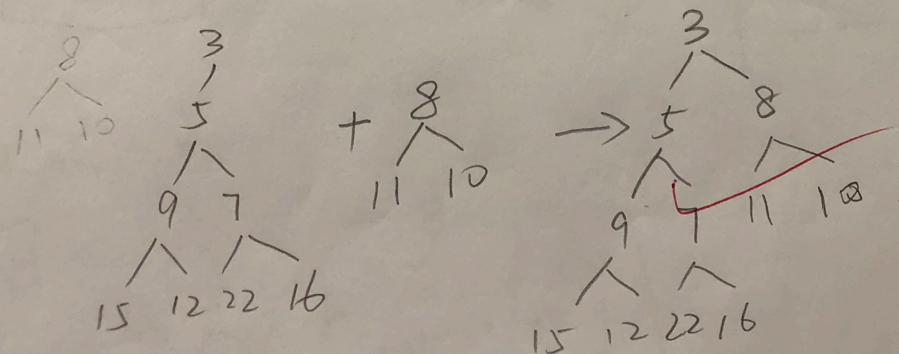
3. (12) Given a set S of 12 records with priorities {15, 9, 12, 5, 22, 7, 16, 3, 2, 8, 11, 10}. Construct a minMax heap for S by using the O(n) bottom-up build heap operations. When done, perform deleteMin once. *Remark:* You must show your tree clearly after each heapify operation for credit.



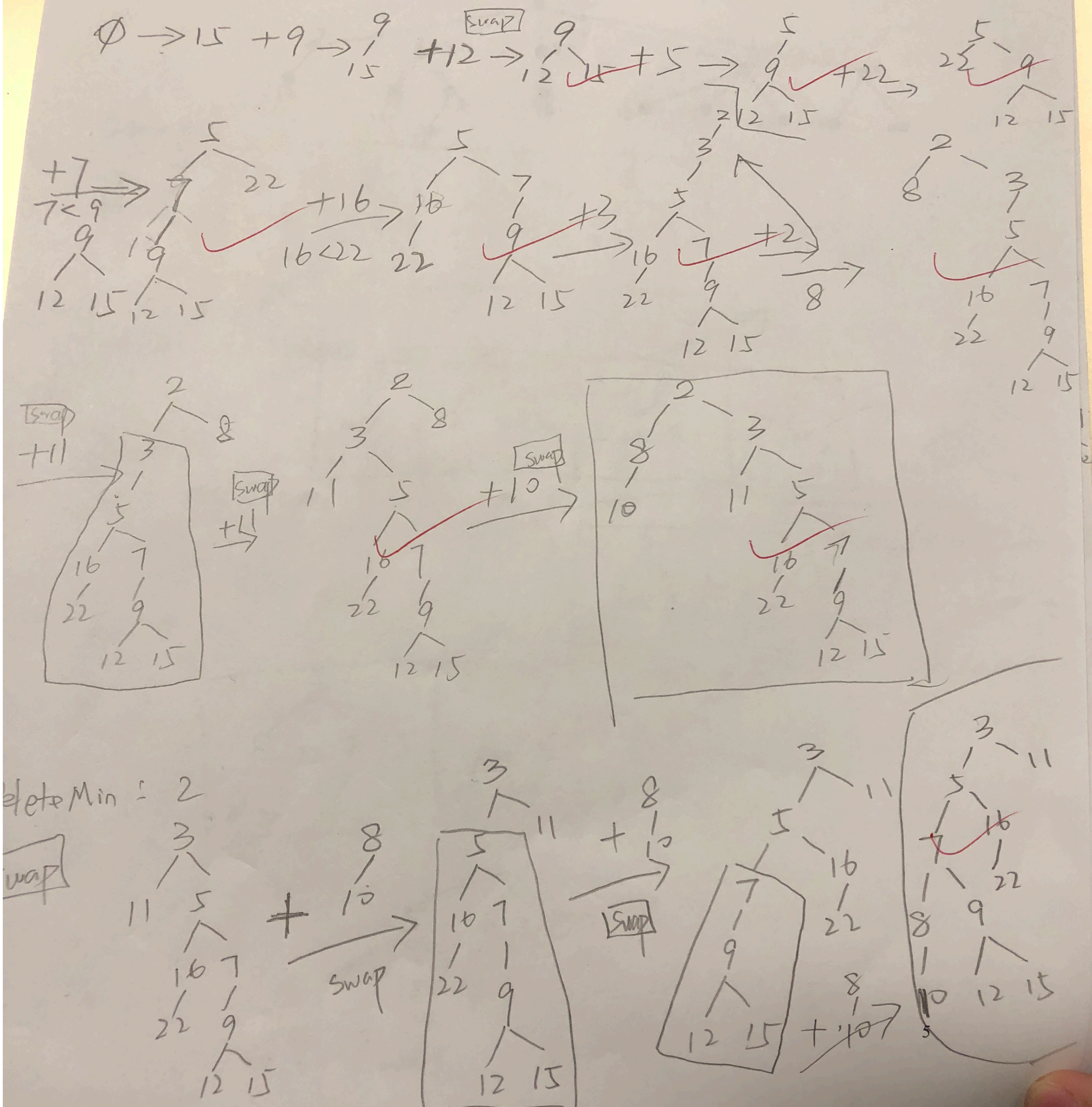
4. (12) Given a set S of 12 records with priorities {15, 9, 12, 5, 22, 7, 16, 3, 2, 8, 11, 10}. Construct a min leftist heap for S by inserting the records in S, in the order given, into an initially empty heap. When done, perform deleteMin once. *Remark:* You must show your tree clearly after each insertion/deletion for credit.



Delete Min 2:

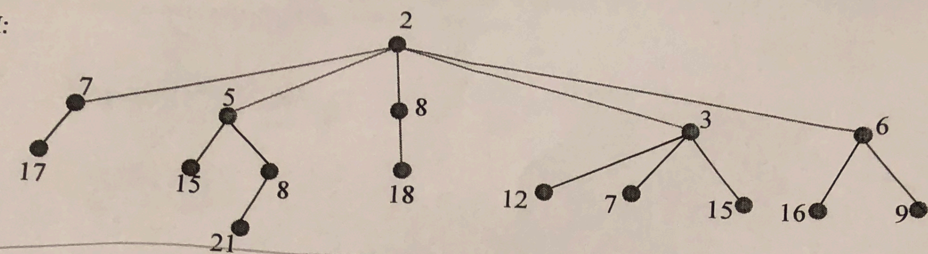


5. (12) Given a set S of 12 records with priorities {15, 9, 12, 5, 22, 7, 16, 3, 2, 8, 11, 10}. Construct a min skew heap for S by inserting the records in S, in the order given, into an initially empty heap. When done, perform deleteMin once. *Remark:* You must show your tree clearly after each insertion/deletion for credit.

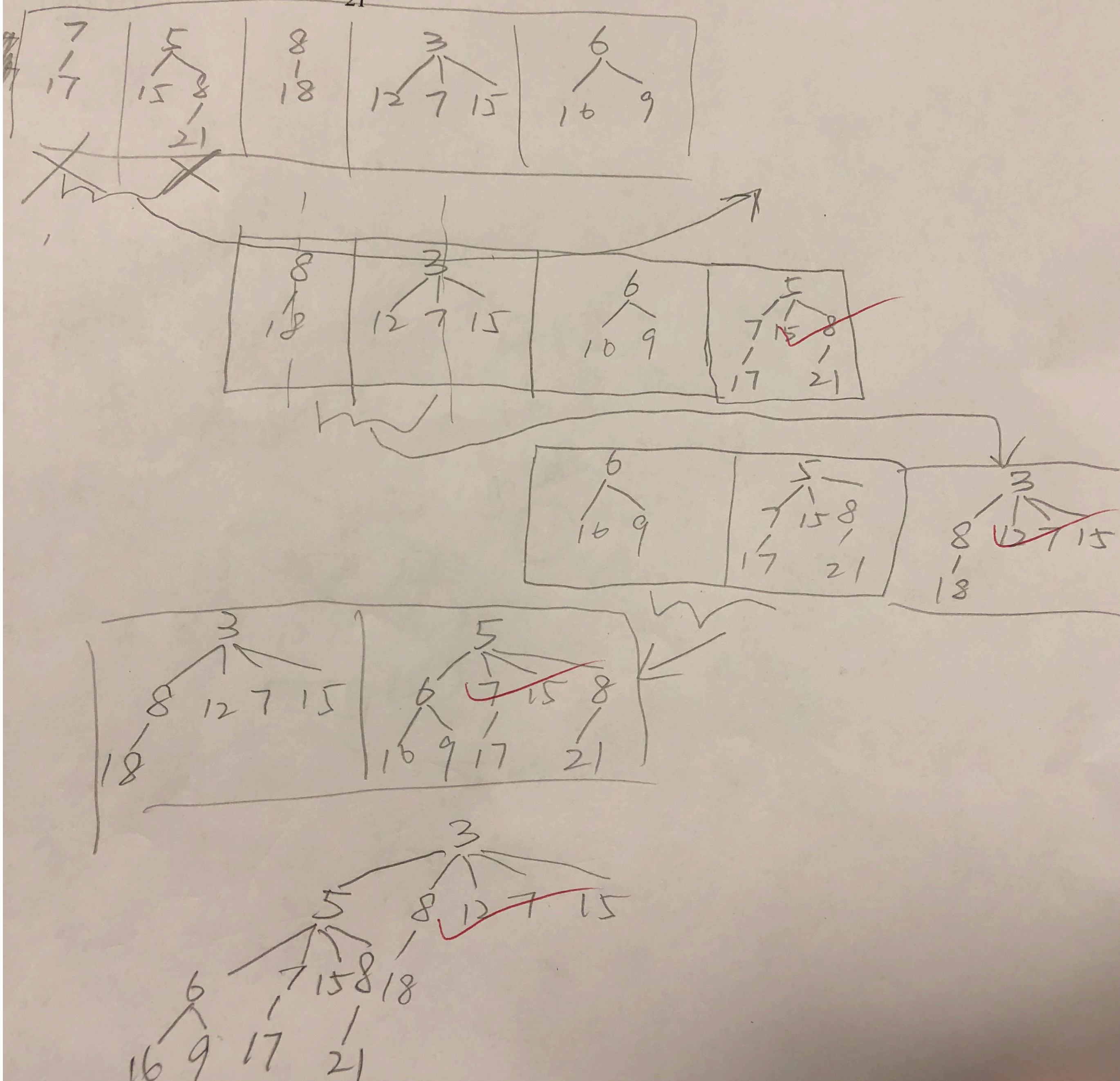


6. (10) Perform deleteMin on the following pairing heap H by using the multi-pass method for merging. *Remark:* You must show your trees clearly after each merging operation for credit.

H:

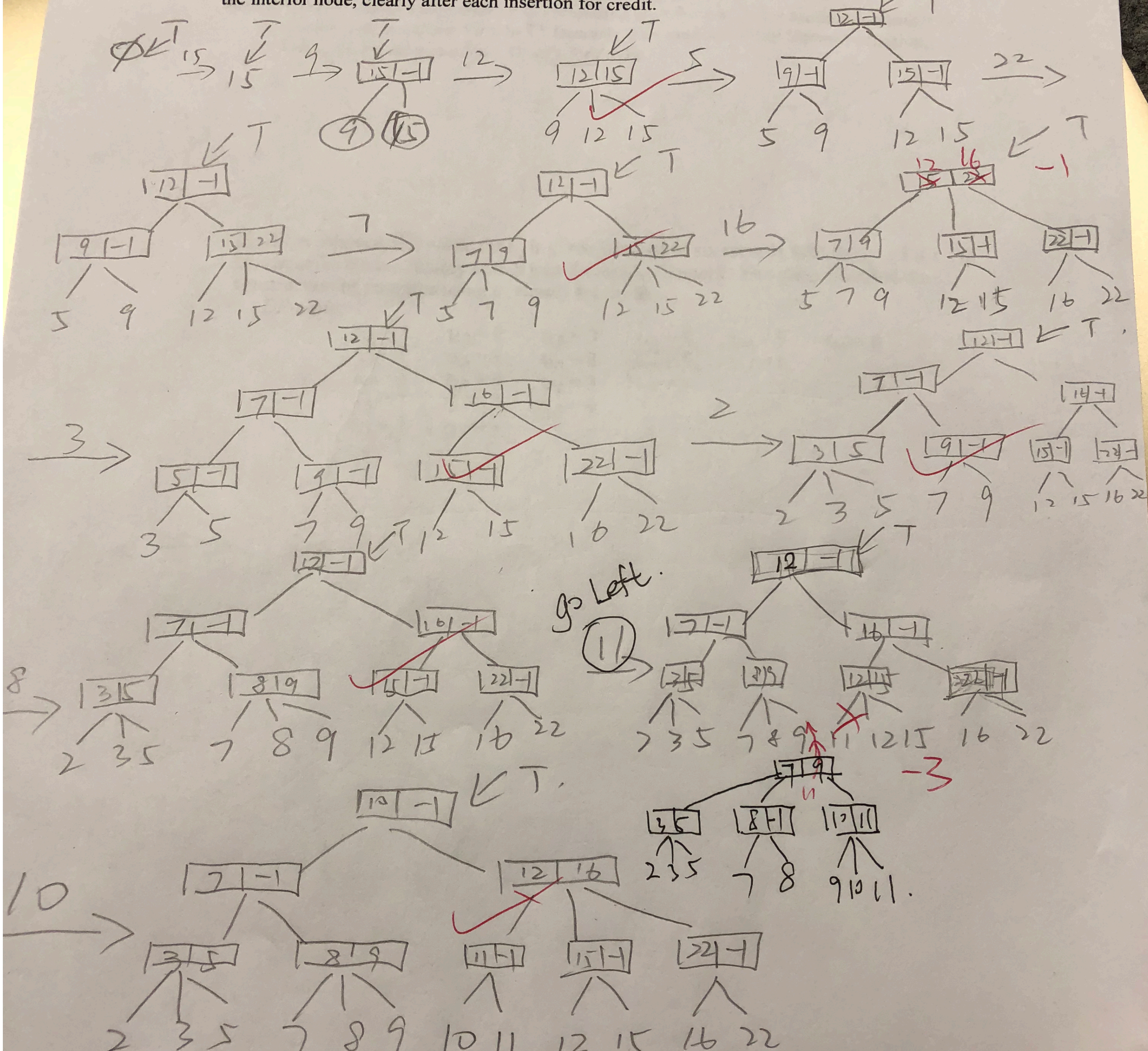


DeleteMin



~~☆~~ (treap). ☆ $\boxed{\text{Search}}$ → Insert

7. (12) Given a set S of 12 records with priorities {15, 9, 12, 5, 22, 7, 16, 3, 2, 8, 11, 10}. Construct a 2-3 tree for S by inserting the given records, in the order given, into an initially empty 2-3 tree. *Remark:* You must show your tree, together with the required information for the interior node, clearly after each insertion for credit.



8. Given a set S of n records with keys x_i, $x_1 < x_2 < \ldots < x_n$, a key x, and the probability function $Pr(x = x_i) = p_i$, $1 \le i \le n$. If dynamic programming is used to construct an optimal binary search tree T for S as discussed in class, answer the following questions.

4

(a) (5) What is the forward equation in computing $c_{i,j}$, where $c_{i,j}$ is the minimum number of comparisons in searching for x in T? *Remark:* You must show the forward equation, including the initial conditions, clearly for credit.

$C_{11} = P_1$

$C_{22} = P_2$

$C_{33} = P_3$

$C_{44} = P_4$

$C_{ii} = P_i$

$$C_{ij} = \min\{C_{i,k-1} + C_{k+1,j}\} + \sum_{L=i}^{j} P_L$$

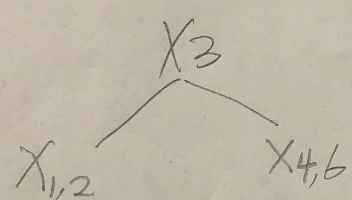
$C_{i+1,i} = 0$

(b) (7) If the following DP-table for $t_{i,j}$ is given for a set of six records with keys x_i, $1 \le i \le 6$, construct the optimal binary search tree T using $t_{i,j}$. *Remark:* You must illustrate the construction of your tree using $t_{i,j}$ clearly for credit.

DP-table:

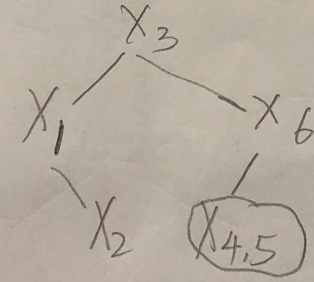
$t_{1,1} = 1$	$t_{2,2} = 2$	$t_{3,3} = 3$	$t_{4,4} = 4$	$t_{5,5} = 5$	$t_{6,6} = 6$
$\boxed{t_{1,2} = 1}$	$t_{2,3} = 3$	$t_{3,4} = 3$	$\boxed{t_{4,5} = 5}$	$t_{5,6} = 6$	
$t_{1,3} = 2$	$t_{2,4} = 3$	$t_{3,5} = 3$	$\boxed{t_{4,6} = 6}$		
$t_{1,4} = 3$	$t_{2,5} = 3$	$t_{3,6} = 4$			
$t_{1,5} = 3$	$t_{2,6} = 3$				
$t_{1,6} = 3$					

$t_{1,6} = 3$



$t_{4,6} = 6$

$t_{1,2} = 1$



$t_{4,5} = 5$