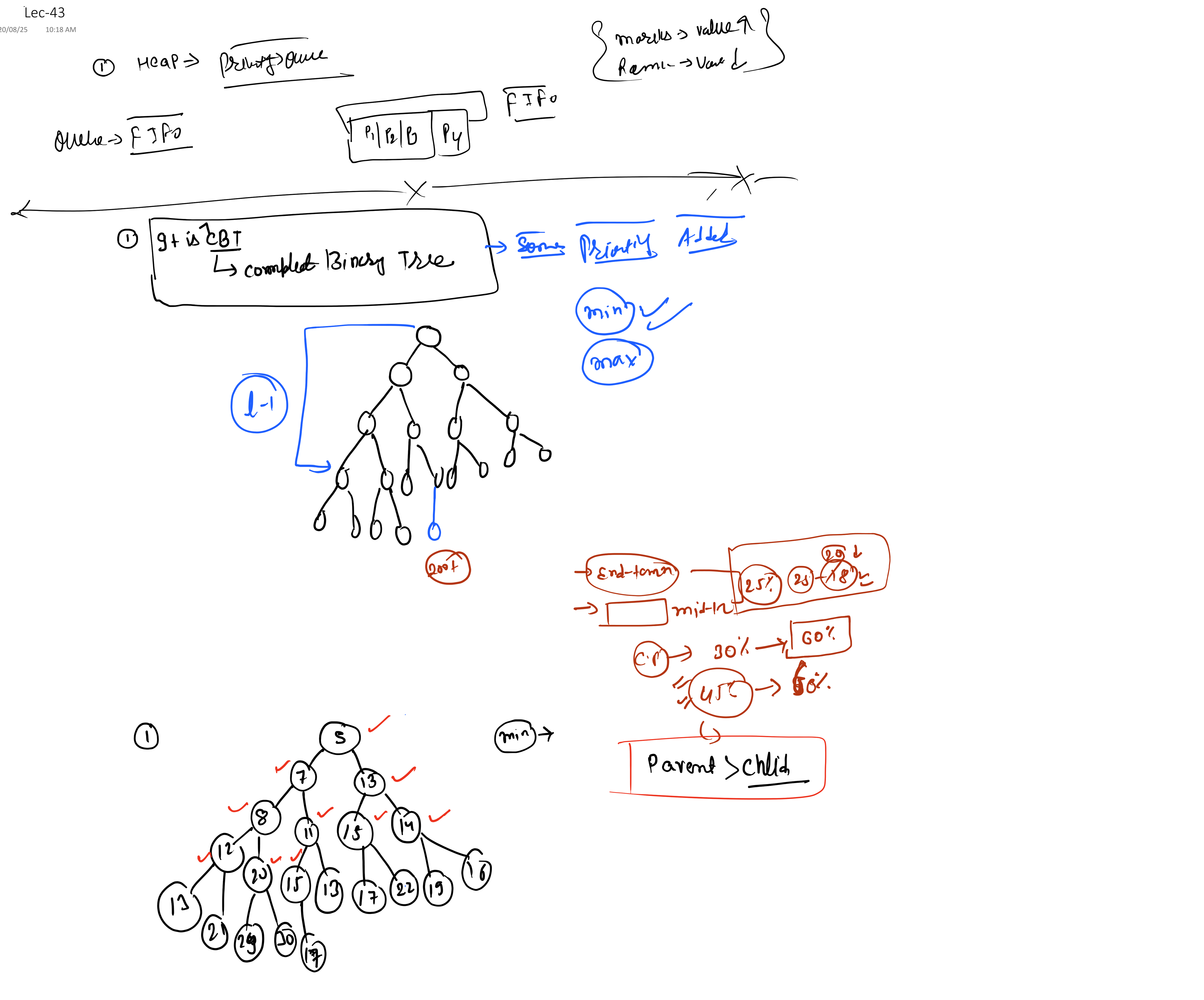
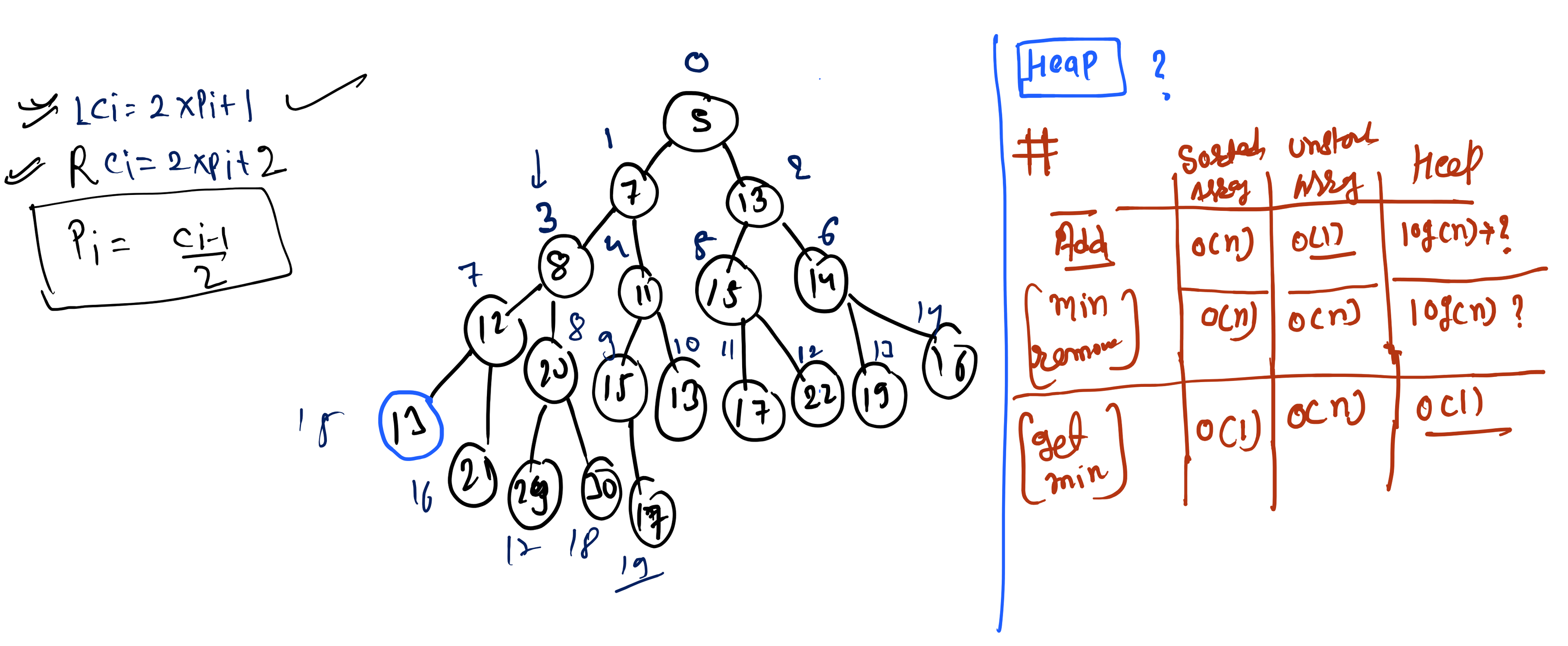
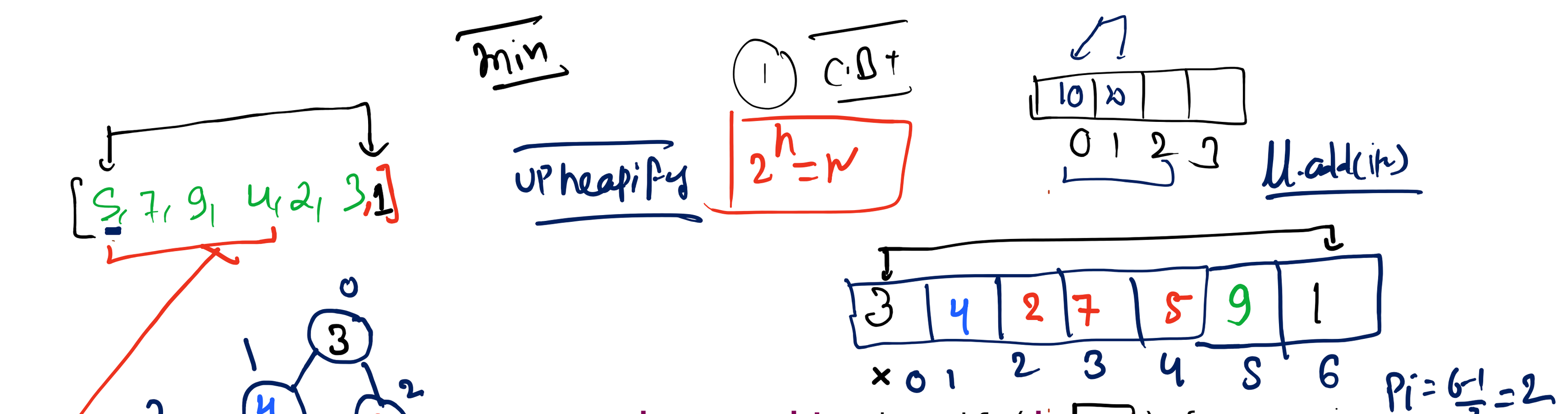
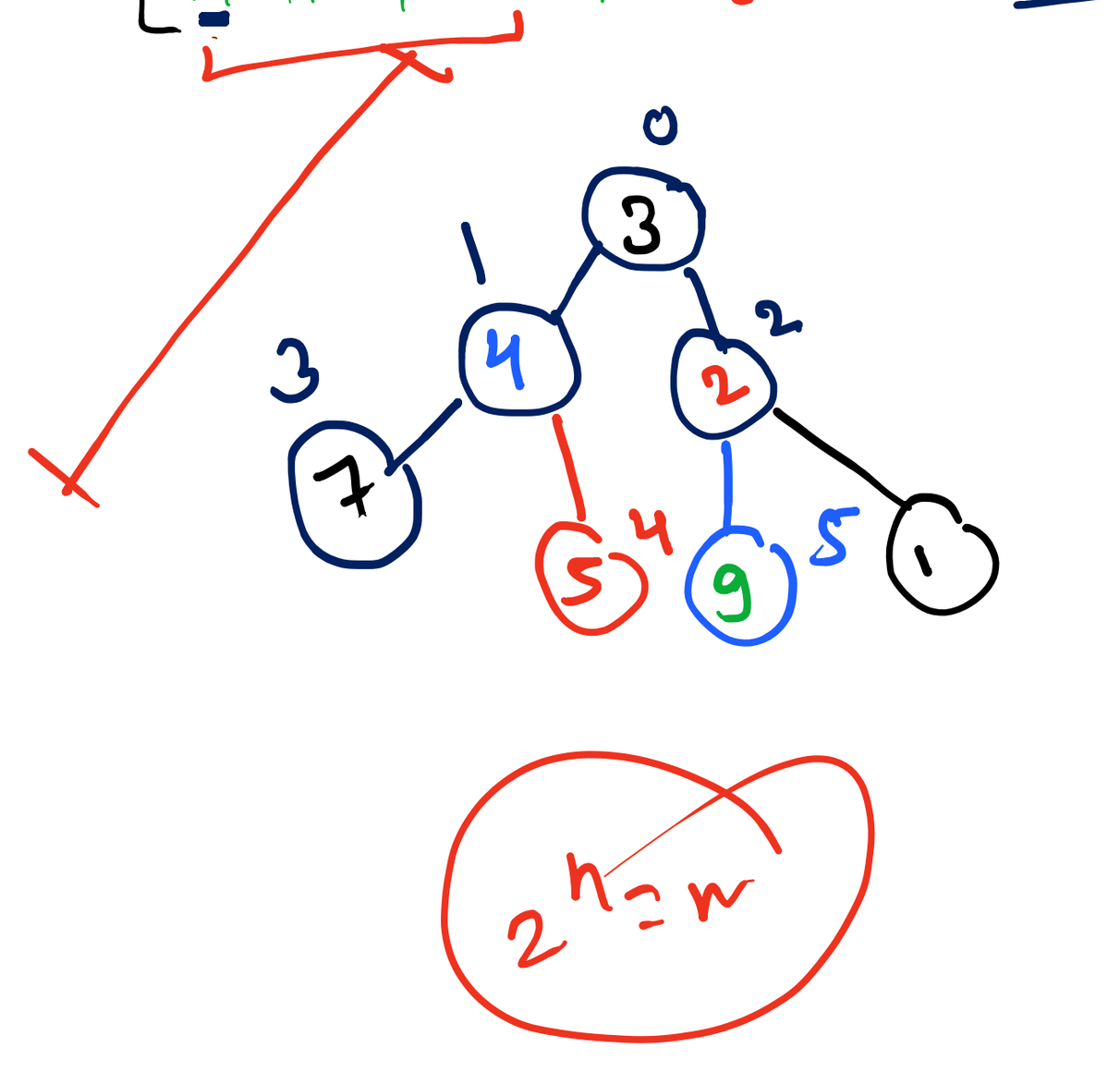
Lec-43

20/08/25 10:18 AM

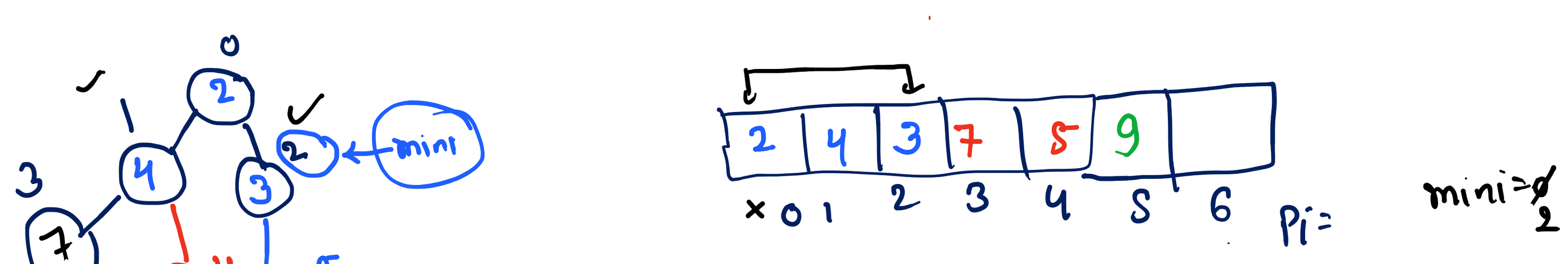


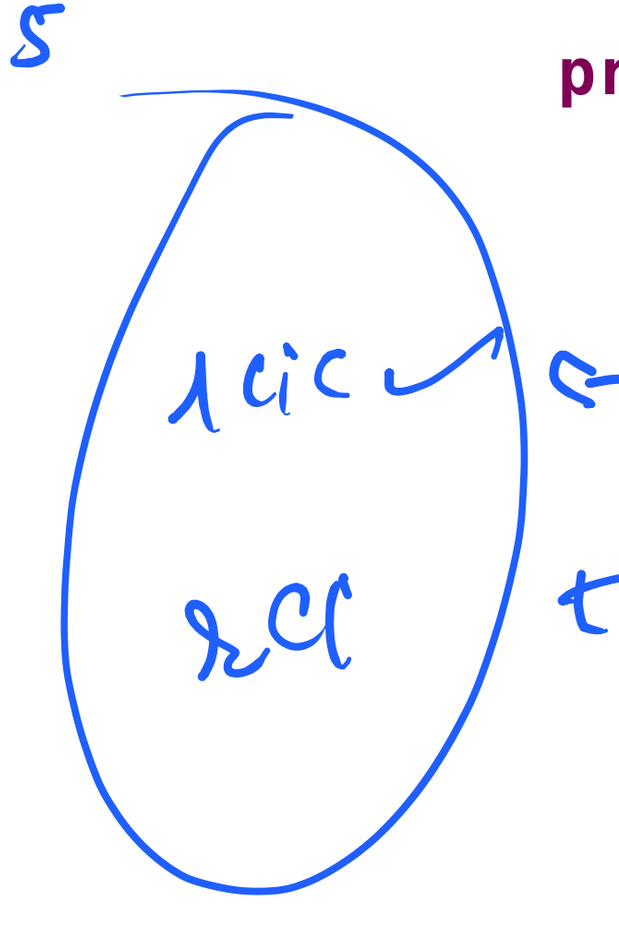




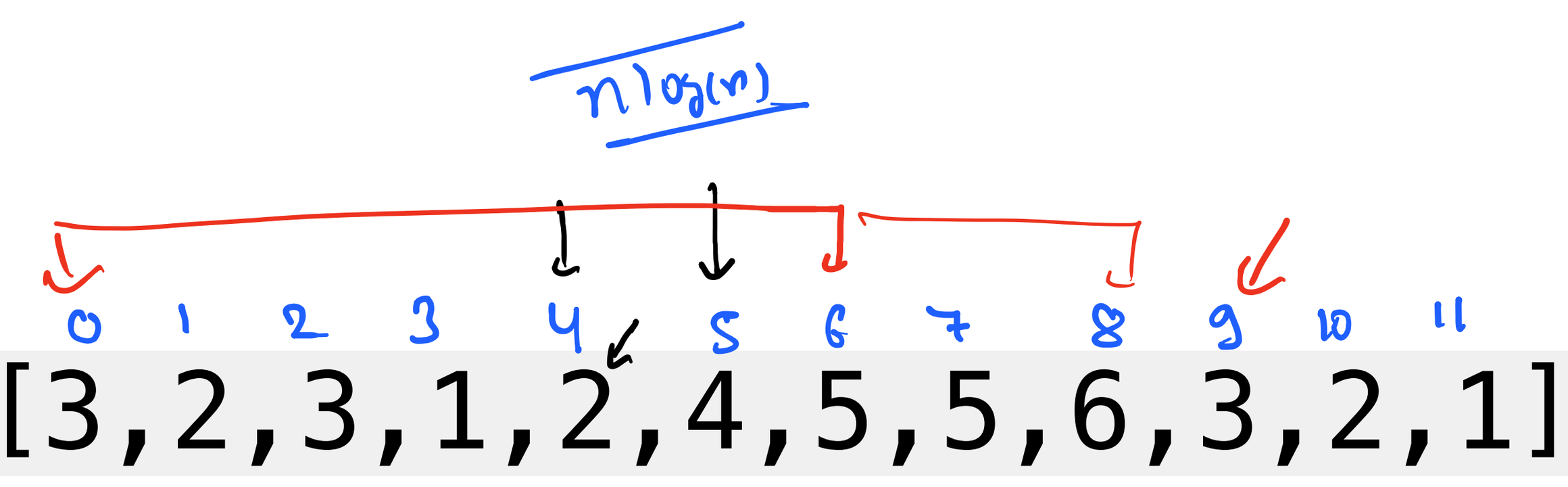


```java
private void upheapify(int ci) {
	// TODO Auto-generated method stub
	int pi=(ci-1)/2;
	if(ll.get(pi)>ll.get(ci)) {
		swap(pi,ci);
	}
}
```



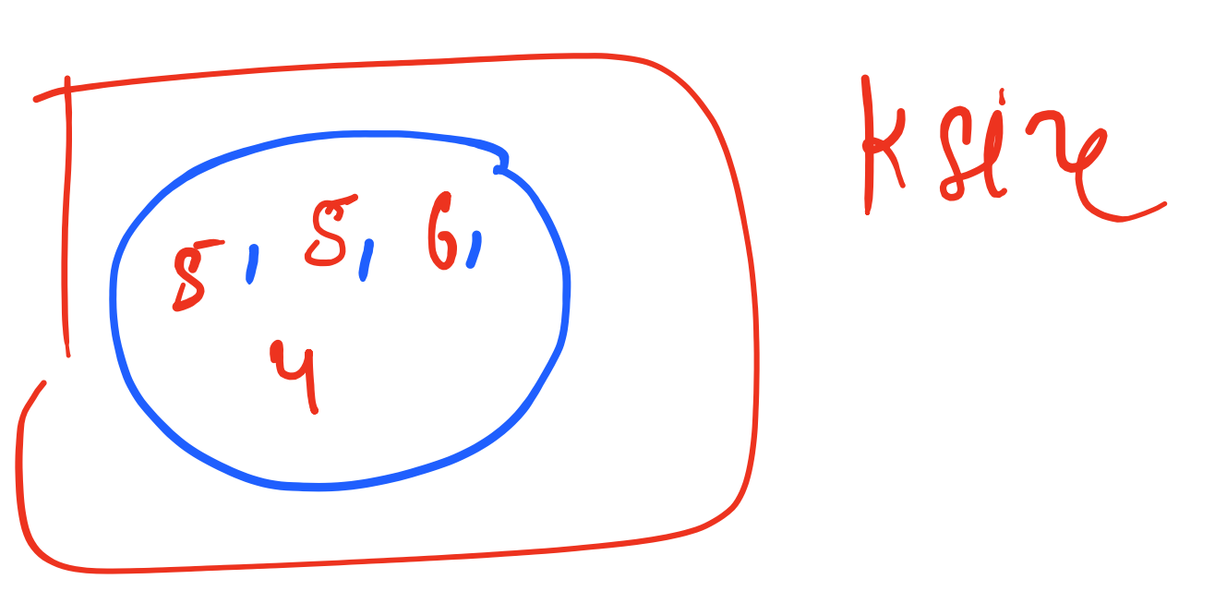


```java
private void downheapif(int pi) {
	// TODO Auto-generated method stub
	int lci=2*pi+1;
	int rci=2*pi+2;
	int mini=pi;
	if(ll.get(mini)>ll.get(lci)) {
		mini=lci;
	}
	if(ll.get(mini)>ll.get(rci)) {
		mini=rci;
	}
	if(mini!=pi) {
		swap(mini, pi);
	}
}
```

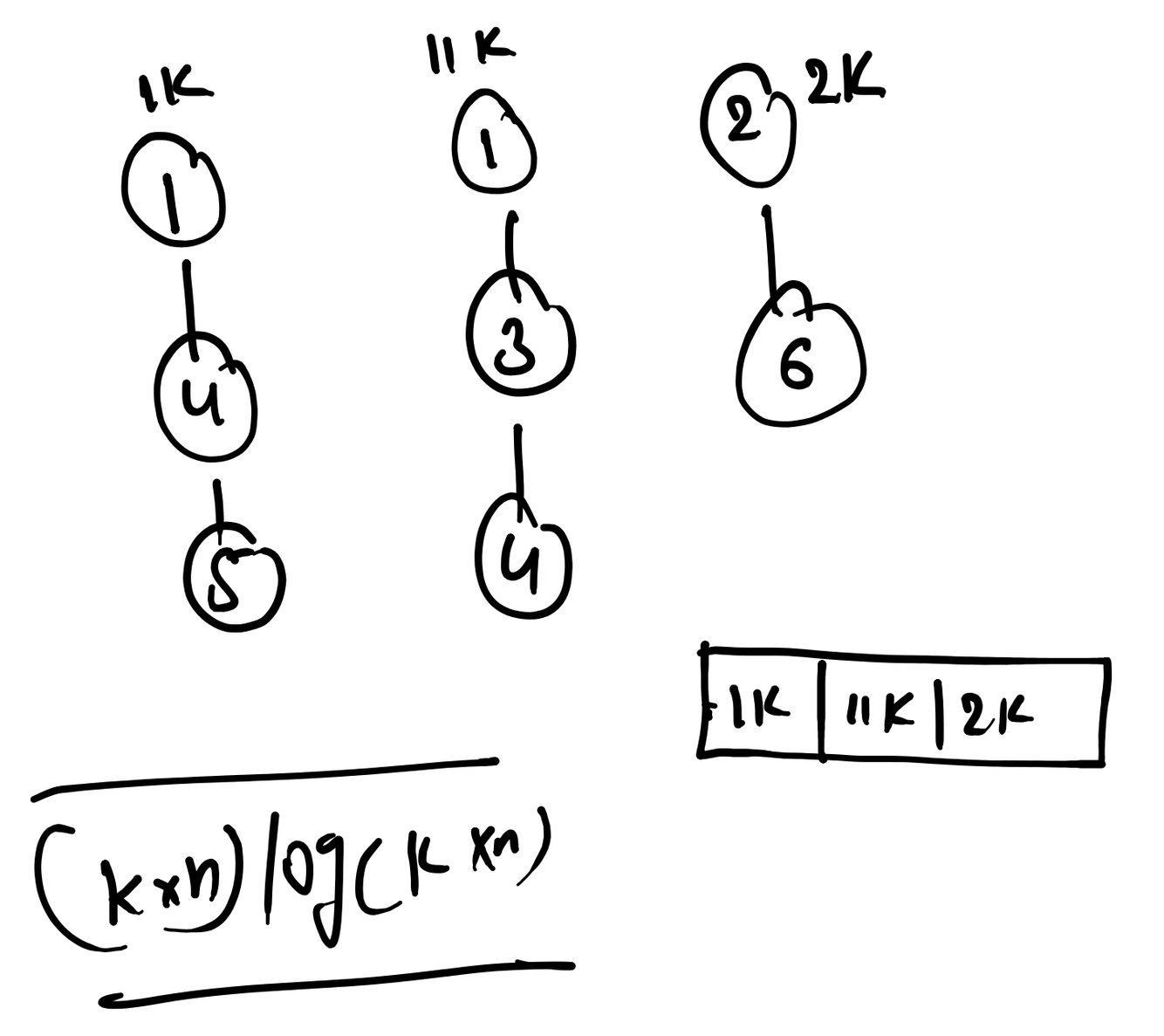


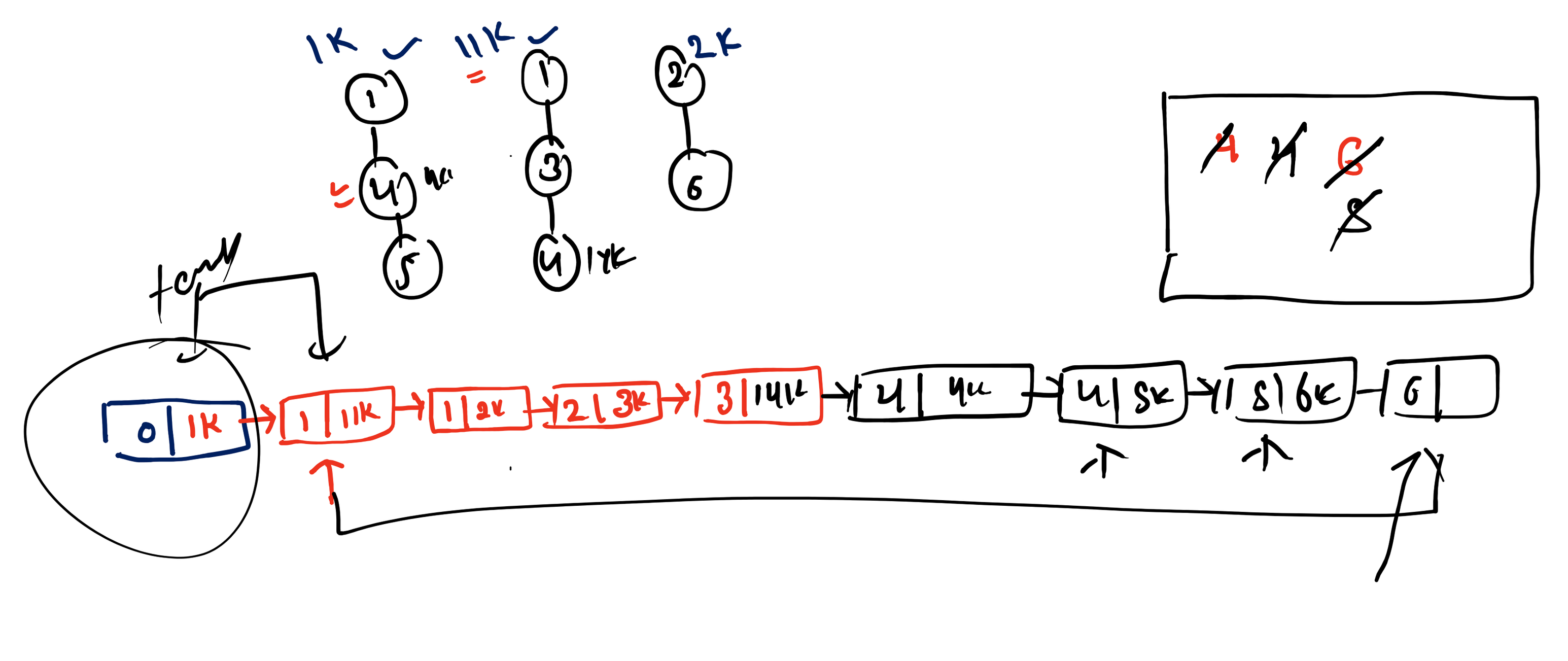
Input: nums = [3,2,3,1,2,4,5,5,6,3,2,1], k = 4

Output: 4



```
Input: lists = [[1,4,5],[1,3,4],[2,6]]
Output: [1,1,2,3,4,4,5,6]
Explanation: The linked-lists are:
[
  1->4->5,
  1->3->4,
  2->6
]
merging them into one sorted linked list:
1->1->2->3->4->4->5->6
```





```java
public ListNode mergeKLists(ListNode[] lists) {
		PriorityQueue<ListNode> pq = new PriorityQueue<>();
		ListNode Dummy= new ListNode();
		ListNode temp=Dummy;
		for (int i = 0; i < lists.length; i++) {
			if(lists[i]!=null) {
				pq.add(lists[i]);
			}
		}
		while(!pq.isEmpty()) {
			ListNode r=pq.poll();
			Dummy.next=r;
			Dummy=Dummy.next;
			if(r.next!=null) {
				pq.add(r.next);
			}
		}
}
```