

### Extra Credit 3.

---

**Question 1.** Given  $k$  sorted arrays of size  $n$ , design an algorithm merge these arrays into a single sorted array of size  $kn$ . (Hint: A sorted array is a min heap. If you have  $k$  min heaps, can you find and delete the smallest number in each of them? If so, can you create another min heap with your  $k$  smallest numbers? In the new min heap with  $k$  numbers, can you sort the numbers? The time complexity of your solution should be  $O(nk \log k)$ .)

Example:

$a1 = [0, 2, 4, 6],$

$a2 = [1, 5, 8, 10],$

$a3 = [3, 7, 9, 20],$

Output: Merged array: [ 0 1 2 3 5 6 7 8 9 10 20]

**Question 2.** Sort a given random array of  $n$  numbers where each element is at most  $k$  index away from its position in the sorted array. (Hint: You could make min/max heaps using the first  $k+1$  numbers and then 1- delete the root *once* 2- add the next element to the heap 3-Repeat 1 and 2 until you cover all the elements. The time complexity of your solution should be  $O(n \log k)$ .)

Example 1: Input:  $a = [2, 8, 0, 17, 5, 12]$  ,  $k = 2,$

(Hint: a number at index 4 in our sorted array, can be located in 2, 3, 4, 5, 6 indices in the given array.)

Output: [0 2 5 8 12 17]