

1 / 1  
points

1.

**Nuts and bolts.** A disorganized carpenter has a mixed pile of  $n$  nuts and  $n$  bolts. The goal is to find the corresponding pairs of nuts and bolts. Each nut fits exactly one bolt and each bolt fits exactly one nut. By fitting a nut and a bolt together, the carpenter can see which one is bigger (but the carpenter cannot compare two nuts or two bolts directly). Design an algorithm for the problem that uses  $n \log n$  compares (probabilistically).

*Note: these interview questions are ungraded and purely for your own enrichment. To get a hint, submit a solution.*

d

Your answer cannot be more than 10000 characters.

**Thank you for your response.**

*Hint:* modify the quicksort partitioning part of quicksort.

*Remark:* This research paper gives an algorithm that runs in  $n \log^4 n$  time in the worst case.

1 / 1  
points

2.

**Selection in two sorted arrays.** Given two sorted arrays  $a[ ]$  and  $b[ ]$ , of sizes  $n_1$  and  $n_2$ , respectively, design an algorithm to find the  $k^{th}$  largest key. The order of growth of the worst case running time of your algorithm should be  $\log n$ , where  $n = n_1 + n_2$ .

- Version 1:  $n_1 = n_2$  and  $k = n/2$
- Version 2:  $k = n/2$
- Version 3: no restrictions

d

Your answer cannot be more than 10000 characters.

**Thank you for your response.**

*Hint:* there are two basic approaches.

- Approach A: Compute the median in  $a[ ]$  and the median in  $b[ ]$ .  
Recur in a subproblem of roughly half the size.
- Approach B: Design a constant-time algorithm to determine whether  $a[i]$  is the  $k^{th}$  largest key. Use this subroutine and binary search.

Dealing with corner cases can be tricky.



1 / 1  
points

3.

**Decimal dominants.** Given an array with  $n$  keys, design an algorithm to find all values that occur more than  $n/10$  times. The expected running time of your algorithm should be linear.

d

Your answer cannot be more than 10000 characters.

**Thank you for your response.**

*Hint:* determine the  $(n/10)^{th}$  largest key using quickselect and check if it occurs more than  $n/10$  times.

*Alternate solution hint:* use 9 counters.

