Interview Questions: Analysis of Algorithms (ungraded) | Coursera

Interview Questions: Analysis of Algorithms (ungraded)

Total points 3

1.

Question 1

3-SUM in quadratic time. Design an algorithm for the 3-SUM problem that takes time proportional to n2n^2n2 in the worst case. You may assume that you can sort the nnn integers in time proportional to n2n^2n2 or better.

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

1 / 1 point

```
public List<List<Integer>> threeSum(int[] nums) {        List<List<Integer>> triplets = new ArrayList<>();        Arrays.sort(nums);        for(int i = 0; i < nums.length; i++) {            for(int j = i + 1; j < nums.length; j++) {                int k = Arrays.binarySearch(nums, (nums[i] + nums[j]) * -1);                if (k >= 0 && k != i && k != j){                    List<Integer> triplet = new ArrayList<>(List.of(nums[i], nums[j], nums[k]));                    Collections.sort(triplet);                    if (!triplets.contains(triplet)) {                        triplets.add(triplet);                    }                }            }        }        return triplets;    }
```

✓

Correct

Hint: given an integer xmathtt{x}x and a sorted array a[]mathtt{a[ ]}a[] of nnn distinct integers, design a linear-time algorithm to determine if there exists two distinct indices imathtt{i}i and jmathtt{j}j such that a[i]+a[j]==xmathtt{a[i] + a[j] == x}a[i]+a[j]==x.

2.

Question 2

Search in a bitonic array. An array is *bitonic* if it is comprised of an increasing sequence of integers followed immediately by a decreasing sequence of integers. Write a program that, given a bitonic array of nnn distinct integer values, determines whether a given integer is in the array.

- Standard version: Use ~3lgnsim 3 lg n~3lgn compares in the worst case.
- Signing bonus: Use ~2lgnsim 2 lg n~2lgn compares in the worst case (and prove that no algorithm can guarantee to perform fewer than ~2lgnsim 2 lg n~2lgn compares in the worst case).

1 / 1 point

a=Find longest increasing subsequence x=Find longest decreasing subsequence ans=number of elements in vector - (a+x-1) a+x-1 = gives maximum length of bitonic array we want to find minimum nmumber of removals , so we can subtract maximum length from the size of the vector to get the number of removals.

Correct

Hints: Standard version. First, find the maximum integer using ~1lgnsim 1 lg n~1lgn compares—this divides the array into the increasing and decreasing pieces.

Signing bonus. Do it without finding the maximum integer.

3.

Question 3

Egg drop. Suppose that you have an nnn-story building (with floors 1 through nnn) and plenty of eggs. An egg breaks if it is dropped from floor TTT or higher and does not break otherwise. Your goal is to devise a strategy to determine the value of TTT given the following limitations on the number of eggs and tosses:

- Version 0: 1 egg, ≤Tle T≤T tosses.
- Version 1: ~1lgn sim 1 lg n~1lgn eggs and ~1lgnsim 1 lg n~1lgn tosses.
- Version 2: ~lgTsim lg T~lgT eggs and ~2lgTsim 2 lg T~2lgT tosses.
- Version 3: 222 eggs and ~2nsim 2 sqrt n~2n

---

tosses.

- Version 4: 222 eggs and ≤cTle c sqrt T≤cT

---

 tosses for some fixed constant ccc.

1 / 1 point

One for each floor

✓

Correct

Hints:

- Version 0: sequential search.
- Version 1: binary search.
- Version 2: find an interval containing TTT of size ≤2Tle 2T≤2T, then do binary search.
- Version 3: find an interval of size nsqrt nn

, then do sequential search. Note: can be improved to ~2nsim sqrt{2n}~2n

---

 tosses.
- Version 4: 1+2+3+…+t ~ 12t21 + 2 + 3 + ldots + t ; sim ; frac{1}{2} t^21+2+3+…+t~21t2. Aim for c=22c = 2 sqrt{2}c=22

.