Tech Interview Prep Notes

Blake

Day 1

Q: Networking Interview Questions

As of now, we don't have them scheduled.

- Restate the interview problem
- Don't jump into finding a solution (don't invest your time into one solution without investigating other option)
- Ask "Is this okay," instead of immediately explaining. The beginning: take things into your own hands. Talk from the beginning, Don't jump into coding. Edge cases:

5 mins Talking: Ensure your understanding of the problem. * restate the question * make a test case 30 mins: Develop A solution

Sample Question: Sort a List. * homogenous types? * sort in place or return a copy * time or space more important * what kind of list? Linked List? Array? * duplicates in list? * order? * inplace?

- consider constraints TODO: time complexity of recursive generate all permuations Todo: RedBlack tree Explore types in python

Counting Sort,

Linear time sorting algorithm; Desire sorted array $\vec{x} \in \mathbb{N}^n$, where n is a sufficiently small natural number.

1. Initialize auxiliary array h with the same length as the range of the values in $\vec{x}$.
2. While iterating through each element $x_i \in \vec{x}$, increment h_{x_i} by 1.
3. Reconstruct the sorted vector

Time: Linear in the length of the array to be sorted. Space: Linear in the range of the value of elements in the array to be sorted. Constraint: small number of unique elements, known range of elements, elements not sparse

Pseudo-Code (untested)

```python
def Counting_Sort(arr: List[int]):
    counts = [0] * len(arr)
    for item in arr:
        counts[item]+=1

    # over-write elements
    for i,item in enumerate(counts):
        if item != 0:
            arr.extend([item]*i)
```