

Interview Beat Down

How I am studying





Data Structures

Know the basics - everyone works with lists and maps on a daily basis so know those well. The rest know how they work and tips/tricks to them.

- Trees, Graphs, Heaps
 - Breadth First Search / Depth First Search (BFS, DFS)
- Arrays
 - quick sort, merge sort, bubble sort
- Linked Lists
- Maps



Problem Solving

1. What is the problem
 - a. Find keywords in the sentences of the problems
 - i. (strings, arrays, sorted array, palindrome)
 - ii. Strings & arrays are very similar, a string is an array of characters.
 1. `String temp = 'abc' => temp.charAt(0) => 'a'`
 2. `Array temp = 'abc' => temp[0] => 'a'`
2. Talk about your problems
 - a. Draw, diagram, pseudo code anything to help you work through the problem out loud



Solutions

1. Having clear and concise logic is more important than solving the problem
 - a. Interviewer needs to be able to follow you
 - i. Imagine the interviewer wasn't even there, they just walked in. Can they understand the code?
2. When whiteboarding make sure you leave space for when you finish the problem

Example:

```
Fn doSomething(inptA, inptB):
```

```
    // leave space for error checking
```

```
    If (inptA === null || inputA === undefined) return null; // etc... you may forget these at the start
```

```
    // talk to interviewer about assumptions
```



Correctness

It is more important to write clean beautiful code than getting the optimal solution. Optimization is always done after coding is completed.

Example: given two strings find if they are a permutation of one another

- Input: 'abc', 'bca' => true
- Input: 'def', 'jkc' => false

function isPermutation(strA, strB): => return true/false if strings are a permutation

1. return strA.sort() === strB.sort();

VS:

1. strA => quickSort
2. strB => quickSort
3. Loop through strings and compare each character at index