Carnegie Mellon University
Silicon Valley

Technical Interview Workshop

Yongwhan Lim
Research Software Engineer
@ Google Research Machine Learning

2pm PT, Thursday, February 25, 2020

Background

- **Yongwhan Lim**
- Research Software Engineer
- Google Research Machine Learning
- http://yongwhan.github.io/
- Stanford
  - CS (BS ‘11 & MS ‘12)
  - Mathematics (BS ‘11)
- MIT: Operations Research (PhD, on extended leave)

Overview

- **Part I**
  - Interview Types
  - Technical Interview
  - Interview Topics
  - 3 Sample Interview Questions
  - Interview Preparation Resources
- **Part II**: Q&A

Part I

Interview Types

- Technical Interview
  - tests technical skill-sets required for a job.
- Behavioral Interview
  - tests soft skills (e.g., effective communication, conflict resolution, etc.)

Interview Types

- **Technical Interview**
  - tests technical skill-sets required for a job.
- Behavioral Interview
  - tests soft skills (e.g., effective communication, conflict resolution, etc.)

Technical Interview

Overview (Company Dependent)

- Recruiter Call
- 0-1 Online Coding Challenge
  - automated screening with 2-3 questions.
- 2-3 Technical Phone Screens
  - first technical conversation with human.
- 4-7 Interviews in Onsite
  - similar to phone screening but more in-depth; you may get probed on your claimed expertise.
- 0-5 Fit Calls & Negotiation

Technical Interview

Overview (Company Dependent)

- Recruiter Call
- **0-1 Online Coding Challenge**
  - automated screening with 2-3 questions.
- **2-3 Technical Phone Screens**
  - first technical conversation with human.
- **4-7 Interviews in Onsite**
  - similar to phone screening but more in-depth; you may get probed on your claimed expertise.
- 0-5 Fit Calls & Negotiation

Interview Topics

Overview

- Data Structures and Algorithms
- (> entry level) System Design Problems

Interview Topics

Overview

- **Data Structures and Algorithms**
- (> entry level) System Design Problems

Interview Topics

Overview

Fundamentals

- Primitive Types
- Arrays & Linked Lists
- Binary Trees
- Heaps
- Sorting

Important

- **Stacks & Queues**
- **Hash Tables**
- **Binary Search Trees**
- **Searching**
- **Recursion**

Real Differentiators

- **Strings**
- **Dynamic Programming**
- **Greedy Algorithms and Invariants**
- **Graphs**

Sample Interview Question #1 (Medium)

Data Structures and Algorithm

Problem Statement (LeetCode #1201)

Given three integers a, b, and c, find n-th smallest positive integer divisible by a or b or c (Note: it is ‘**or**’ not ‘and’).

Sample Interview Question #1 (Medium)

Data Structures and Algorithm

Problem Statement (LeetCode #1201)
Given three integers a, b, and c, find n-th smallest positive integer divisible by a or b or c (Note: it is ‘**or**’ not ‘and’).

Constraints
n, a, b, and c are all at most 1 billion.

Sample Interview Question #1 (Medium)

Data Structures and Algorithm

Problem Statement (LeetCode #1201)
Given three integers a, b, and c, find n-th smallest positive integer divisible by a or b or c (Note: it is ‘**or**’ not ‘and’).

Constraints
n, a, b, and c are all at most 1 billion.

Any Ideas?

Sample Interview Question #1 (Medium)

Data Structures and Algorithm

Binary Search Solution (Logarithmic):

Now, do you see it?

Sample Interview Question #1 (Medium)

Data Structures and Algorithm

Binary Search Solution (Logarithmic):

```
#include<bits/stdc++.h>
using namespace std;

int nthUglyNumber(int n, int a, int b, int c) {
        int low = 1, high = INT_MAX;
        while(low < high) {
                int mid = low + ((high - low) >> 1);
                if(eval(mid, a, b, c) >= n) {
                        high = mid;
                } else {
                        low = mid + 1;
                }
        }
        return low;
}
```

```
typedef long long ll;

ll lcm(ll a, ll b) {
        return a/__gcd(a,b)*b;
}

ll eval(ll x, ll a, ll b, ll c) {
        return x/a + x/b + x/c - x/lcm(a,b) - x/lcm(a,c) - x/lcm(b,c)
+ x/lcm(a,lcm(b,c));
}
```

Sample Interview Question #2 (Hard)

Data Structures and Algorithm

Problem Statement (LeetCode #1312)

Given a string s, find a minimum number of insertions to make it a palindrome.

Sample Interview Question #2 (Hard)

Data Structures and Algorithm

Problem Statement (LeetCode #1312)

Given a string s, find a minimum number of insertions to make it a palindrome.

Constraints

s can have at most five hundred alphabetic characters.

Sample Interview Question #2 (Hard)

Data Structures and Algorithm

Problem Statement (LeetCode #1312)

Given a string s, find a minimum number of insertions to make it a palindrome.

Constraints

s can have at most five hundred alphabetic characters.

Any Ideas?

Sample Interview Question #2 (Hard)

Data Structures and Algorithm

DP Solution (Quadratic):

Now, do you see it?

Sample Interview Question #2 (Hard)

Data Structures and Algorithm

DP Solution (Quadratic):

```
#include<bits/stdc++.h>
using namespace std;

int minInsertions(string &s) {
        int n = s.size();
        vector<vector<int>> dp(n, vector<int>(n,0));
        for (int i = 1; i < n; i++)
                for (int j = 0, k = i; k < n; j++, k++)
                        dp[j][k] = (s[j]==s[k]) ? dp[j+1][k-1] : min(dp[j][k-1],dp[j+1][k])+1;
        return dp[0][n-1];
}
```

Interview Preparation Resources

Data Structures and Algorithm

- Popular Websites
  - LeetCode
  - CodeForces
  - AtCoder, TopCoder, and CodeChef

Interview Preparation Resources

Data Structures and Algorithm

- Popular Websites
  - **LeetCode**
  - **CodeForces**
  - AtCoder, TopCoder, and CodeChef

Interview Preparation Resources

Data Structures and Algorithm

- Popular Websites
  - **LeetCode**
  - **CodeForces**
  - AtCoder, TopCoder, and CodeChef
- Try to solve all problems from biweekly/weekly LeetCode contest fast.
- Aim to be on division 1 at CodeForces; this will trivialize the technical interview.

Interview Preparation Resources

Data Structures and Algorithm

- References:
  - **Elements of Programming Interview, 2nd**
  - Cracking the Coding Interview
  - Overkill:
    - Competitive Programming 4
    - Guide to Competitive Programming

Part II: Q & A

Questions and Answers (1/15)

- How do you overcome nervousness?

Questions and Answers (2/15)

- Could you provide a live solving of a technical question?

Questions and Answers (3/15)

- Is interview process as an intern different from full-time technical interview? How do you get past the automatic filter?

Questions and Answers (4/15)

- What are topics to prepare, the best way to prepare, and programming language expectations?

Questions and Answers (5/15)

- Will interviewer evaluate applicants' technical knowledge other than coding skills?

Carnegie Mellon University Silicon Valley

Questions and Answers (6/15)

- Are there any specific machine learning and artificial intelligence technical questions that frequently show up in interviews (and that we should prepare for)?

Carnegie Mellon University Silicon Valley

Questions and Answers (7/15)

- If I do not have too much background on a position I am applying for, how do I leave a good impression to interviewer?

Questions and Answers (8/15)

- What are some tips to be successful in the interview process at Google?

Questions and Answers (9/15)

- What are some good questions to ask after the interview?

Questions and Answers (10/15)

- How should I communicate with interviewer during interview?

Questions and Answers (11/15)

- What are the good and bad examples of a technical interview?

Questions and Answers (12/15)

- Is LeetCode enough for preparing the technical interview?

Questions and Answers (13/15)

- What do you do if you do not know how to solve the problem?

Questions and Answers (14/15)

- What technologies do you use at Google Ads?

Questions and Answers (15/15)

- (COVID) How is virtual tech interview different from the in-person interview?

Contact Information

- My LinkedIn profile is **https://www.linkedin.com/in/yongwhan-lim-09855328/**. Feel free to add me there! :)
- Also, if you would like to contact me, email **yongwoods@gmail.com**.
- My personal website is **http://yongwhan.github.io/**.