

# Leetcode 101

Created By	Yu Zhou
Due Date	
Importance	
Lane	Workshops
Status	Session Notes
# Time Block / Hour	
Urgency	

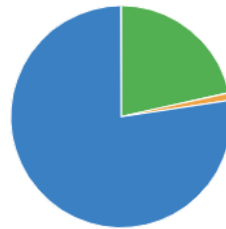
## Outline

- Leetcode now has 1500 Problems

### Your Progress

Session:

2018



1160  
Todo

323 / 1499  
Solved

16  
Attempted

- How many leetcode problems are needed??
  - ▼ Topics
    - 20-30 per topics
    - What topic?

## How to Break Into the Tech Industry-a Guide to Job Hunting and Tech Interviews

I recently completed a job search for my first role as a software engineer. Despite having first learned how to code almost a year before, having a background as an English major and former professional poker player, I was able to land a total of 8 offers including Google, Uber, Yelp, and

<https://haseebq.com/how-to-break-into-tech-job-hunting-and-interviews/#part1>



### ▼ Recommended # of the problems solved

- Tier 1 / FAMNG + Unicorn startup (300-400 Problems):
  - Level: Medium/Hard , Ratio: 60 / 40
  - Lyft / Airbnb / Uber (300-400)
  - Amazon/Microsoft (200-300)
  - Google/Facebook (300-500)
- Tier 2 / Growing Startup + Known large corp.
  - Level: Easy / Medium / Hard , Ratio: 10 / 80 / 10
  - Yahoo / eBay (100 - 200)
  - Intercom / Cisco / Walmart (100 - 200)
- Tier 3 / Tech-related corp + others
  - Level: Easy / Medium / Hard , Ratio: 40 / 55 / 5
  - Synopsys / AMD / Nvidia (100)

### ▼ How to do Leetcode?

#### ▼ Be company specific

##### ▼ Tier 0-1

- Microsoft
- Google
- Pinterest
- Stripe
- Square

- Twitch
- Dropbox
- Apple

- LinkedIn
- Oculus
- Robinhood
- Square
- Affirm
- Nuro
- Quora
- Boston Dynamics

▼ **Tier 2**

- Adobe
- Nvidia
- reddit
- Rubrik
- Tesla
- Slack
- Twitch
- Twitter
- Uber
- SpaceX
- Snap.Inc
- Salesforce
- Yelp
- Atlassian
- Coursera
- Github
- tinder
- Scopely
- Palantir
- Bloomberg
- Didi
- Databricks
- postmates
- Discord

- Skype
- Okta
- Gusto
- VM Ware
- Khan Academy
- addepar
- Asana
- MongoDB
- Mozilla
- coinbase
- figma
- wealthfront
- Spotify - NY
- Zoom
- Tableau
- Audible

▼ **Tier 3**

- Zillow
- NerdWallet
- WalmartLabs
- BlackRock
- Cloudera
- Oath
- Credit Karma
- Box
- Hulu
- Etsy
- Ebay
- Spunk
- pure storage
- Terradata
- viasat

- Intuit
- Intel
- Qualcomm
- Doordash
- Wish
- AutoDesk
- Grubhub
- workday
- Blend
- visa
- Trulia
- Qualia
- ripple
- Instacart
- Argo AI

#### ▼ Trading

- Citadel
- Two Sigma
- addpar
- Jane Street

#### ▼ Gaming

- Epic Games
- Unity
- Niantics
- Ubisoft
- Activision Blizzard
- Square Enix
- Naughty Dog
- Double Fine
- Amazon game studio
- Microsoft studios

- Sony Studios
- EA
- LucasFilm
- Dreamworks
- Valve
- Rockstar
- Bethesda
- Treyarch
- Pocket Gem
- Bungie

#### ▼ Prioritize topics

##### ▼ Scrape Information

- <https://leetcode.com/discuss/interview-question?currentPage=1&orderBy=hot&query=>
- [https://fellows.pathrise.com/static\\_pages/interview\\_prep\\_guide](https://fellows.pathrise.com/static_pages/interview_prep_guide)

##### ▼ Organize data

- ▼ High Frequency
- ▼ Topics

#### ▼ Plan

##### ▼ Keep track of your data

- Kanban / Agile
  - <https://github.com/yuzhoujr/leetcode/projects/4>
- Weekly Plan
- Daily Plan
- Redo

##### ▼ Timer

- ▼ Easy: 15
- ▼ Medium: 25 - 30
- ▼ Hard: 30 - 45

##### ▼ Workflow

##### ▼ **Template for Leetcode**

### **Communication**

☐ **Extract keywords** (*Input constraints, Data types, Scale of input*)

☐ **Classify the Problem**

☐ What are the algorithms commonly used on this kind of problems?

☐ Paraphrase the question

☐ **Understand test cases / Ask for more test cases**

☐ **Function Signature**

☐ **Clarifications / Edge Cases**

## Problem Solving

☐ **Trade Off Analysis**

☐ **Brute Force Solution**

☐ **Outline your approach**

☐ **Space & Time Complexity** \*& \*\*\*(*call stack*) - *specify variables*

(In what situations is the solution good? In what situations is the solution bad?)

☐ **Analyze Inefficiency**

*Bottlenecks, Un-necessary Work, Duplication Work*

☐ **Optimal Solution**

☐ **Outline your approach**

☐ **Space & Time Complexity** \*& \*\*\*(*call stack*) - *specify variables*

☐ **Compare it with Brute Force**

☐ Ask interviewer: "Does this sound like a good approach?"

## Coding / Testing

☐ **Implementation**, keep comparing optimal solution and brute force solution, explain why this solution is better to show your understanding

☐ **Verification**

☐ Scan through

☐ Run test cases - small/medium sized, one more likely to detect errors

▼ Timebox

▼ First Try

▼ If you can't solve within 7 minutes, solution time

- ▼ Second Try
  - ▼ Try not to check solution
    - ▼ Easy: 20 minutes
    - ▼ Medium: 30 minutes
    - ▼ Hard: 40 minutes
  - ▼ Third Try
    - ▼ Easy: 10 minutes
    - ▼ Medium: 20 minutes
    - ▼ Hard: 30 minutes
- ▼ Take a break and study
  - ▼ Discussion Board
  - ▼ YouTube
    - <https://www.youtube.com/watch?v=o8S2bO3pmO4>
    - Study for topics
  - ▼ 9 Chapter
    - ▼ <https://chrome.google.com/webstore/detail/九章刷题小助手/anljhkknafhgofbdhoijjomhnhncblo?hl=zh-CN>
- ▼ Understand why you are stuck
  - Problem solving?
    - No clue at all?
      - Check tag, and study the tag
    - Have clue, but stuck?
      - Check tag, and study the tag
  - Coding?
    - Constantly getting bugs?
      - Formulate a bug check list
    - DFS Error checkIsit
      - ☐ Have I checked my parameter when recursively calling