Table of contents:

[**Date: 28/08/2022**](#_jcl25z4sa3or) **1**

[**Date: Course**](#_6uo6f3t7oeo9) **Setup 1**

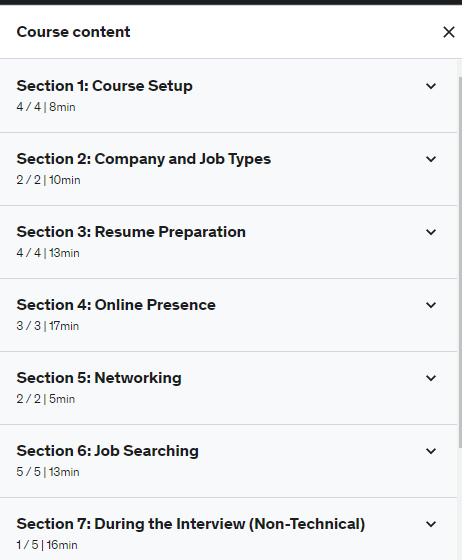
**NOTE 1:** If you are starting a new feature then create a new branch. If you are continuing with a feature then continue with the same branch. Commit the code,and data for your exercise in the branch.

NOTE 2: To update the above table of content, click on the table and select the circle option that appears on the left side

# 29/08/2022: Company and Job Types

Objective or learning goals of this exercise, or what your code/analysis does:

1. Company Types Overview
2. Job types overview



Link to Gitlab repo commit:

# 30/08/2022: Resume preparation

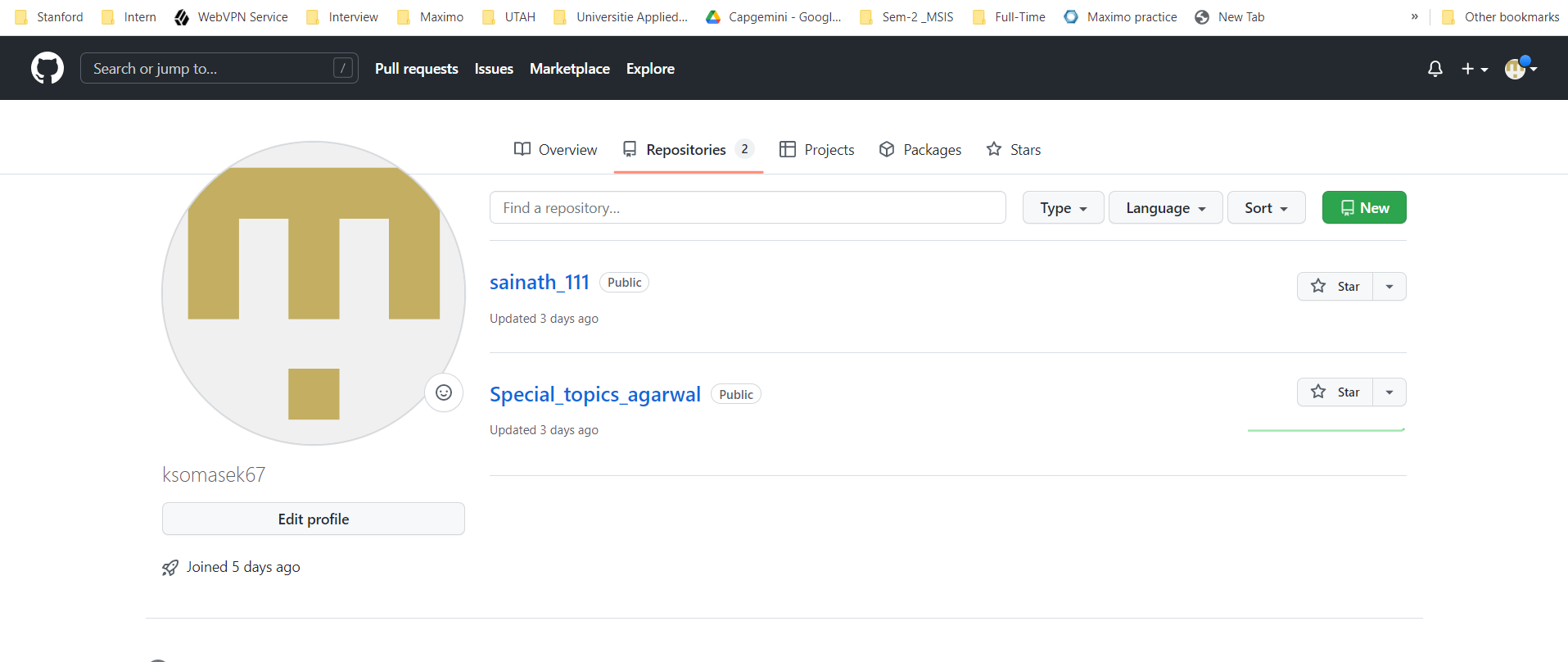
Objective or learning goals of this exercise, or what your code/analysis does:

1. Resume creation tools
2. List of creation websites
3. Resume checklist review

# 01/09/2022: Online Presence

Objective or learning goals of this exercise, or what your code/analysis does:

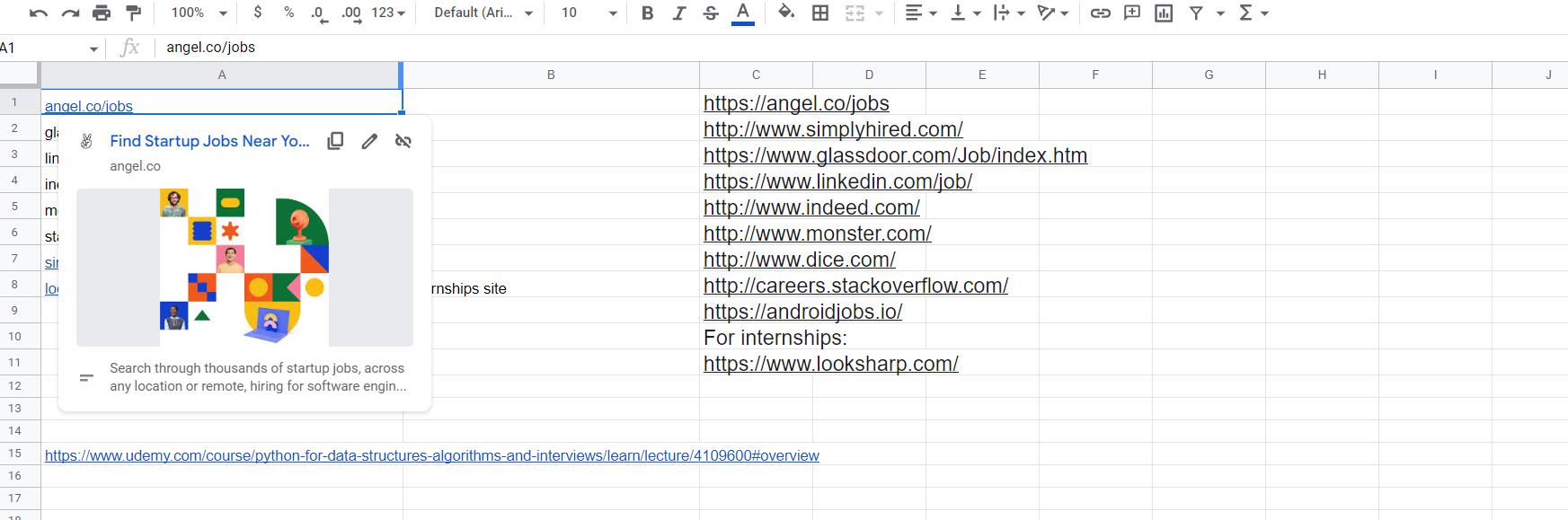
1. Linkedin
2. GITHUB
3. Personal Websites and portfolio

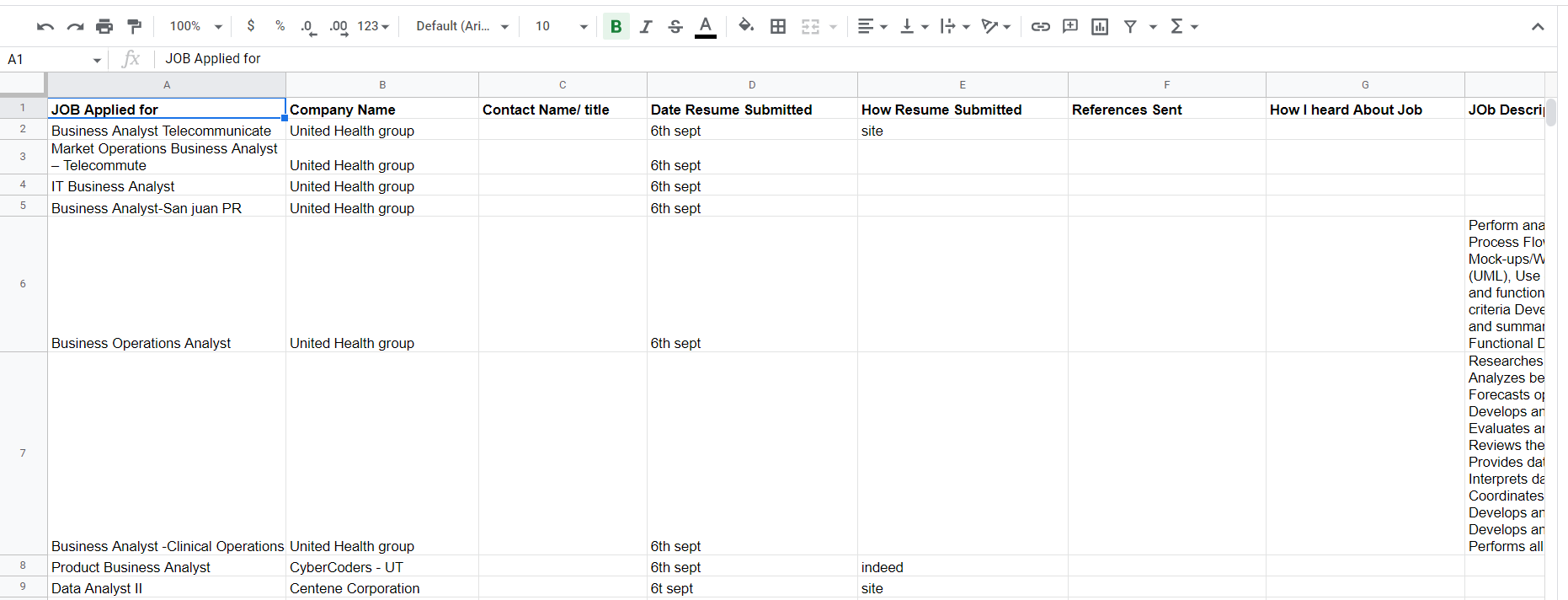


# 02/09/2022: Job Searching

Objective or learning goals of this exercise, or what your code/analysis does:

1. Organization of Job Search
2. Job Search Sites
3. List of Job boards and sites
4. Company matching sites





Link to Gitlab repo commit:

# 05/09/2022: During the Interview

# Overview of interview Section

* Steps of an Interview
* Tough Interview Questions
* Questions for students

**Interview**

Why are you leaving current position?

Higher studies

Why do you want this position?

I wan to achieve my goals, I believe with ABC company I will get a chance to put my knowledge and be a part of ABC’s growth.

Where do you see yourself in 5 years?

As a successful Senior Data Scientist

What is your greatest weakness?

Sometimes I feel frustrated when the work is not done. When I get failed to meet deadlines.

What do you know about your company?

My company is Top 10 MNC consulting company every year providing employment to more than 70000 a year. With revenue increase of 14.6%

What’s the most difficult situation you had to face at a Job?

Lack of support

What do you think makes a successful manager?

Good relationship with Team and addressing and appreciating the Team members for the work Done.

Why did you choose this Profession /major?

To become a Data Scientist, I hope Data Engineer and Data Analyst Roles will help me in becoming a Data Scientist and Data Architect.

Be honest unless you fell into it. In that case, be honest anyway, but show you have an abundance of enthusiasm about being part of the field and are highly motivated to succeed.

List functions:

|  |  |
| --- | --- |
| Operation | BIG-O Efficiency |
| Index[] | O(1) |
| Index assignment | O(1) |
| append | O(1) |
| Pop() | O(1) |
| Pop(i) | O(n) |
| Insert(i,item) | O(n) |
| del operator | O(n) |
| Iteration | O(n) |
| Contains(in) | O(n) |
| get slice[x:y] | O(n) |
| del slice | O(n) |
| set slice | O(n+k) |
| reverse | O(n) |
| concatenate | O(k) |
| sort | O(nlogn) |
| multiply | O(nk) |
|  |  |

Index[] - O(1)

Index assignment

Append

Pop()

Pop(i)

Insert(i,item)

del operator

iteration

contains(in)

get slice[x:y]

# 07/09/2022: Post Interview

# Salary Negotiations

* Salary Questions and Answers
* Preparing References

# 07/09/2022: Technical Break Section

* Technical Overview
* Getting Help for the Technical Section

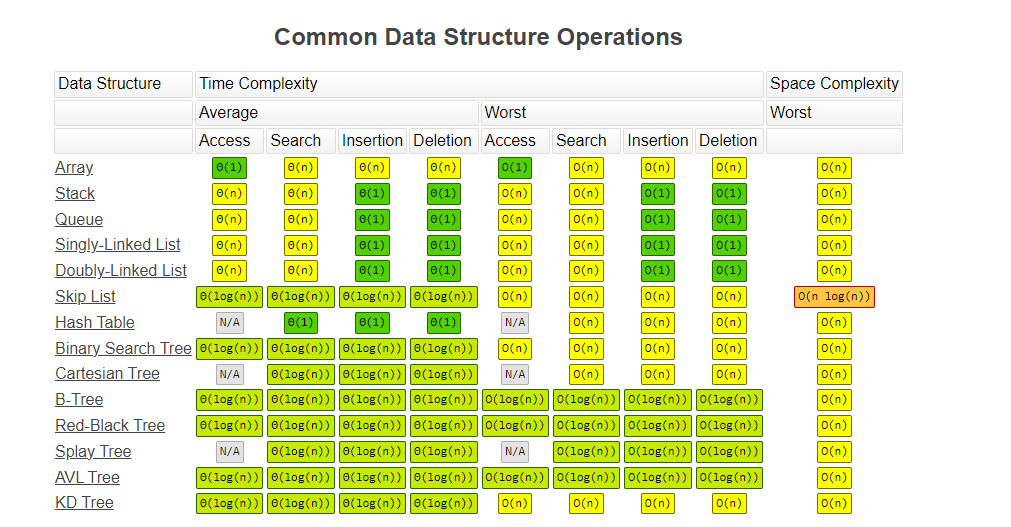
# 07/09/2022: Introduction to jupyter Notebooks

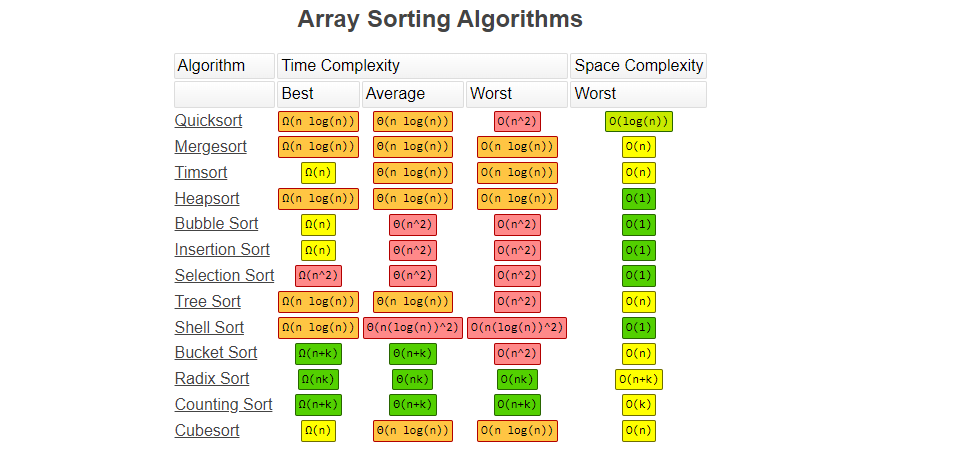
**Dictionaries**:

In python these are an implementation of a Hash table. They operate with keys and Values, for Example:

D={‘k1’:1,’k2’:2}

|  |  |
| --- | --- |
| Operation | Big-O Efficiency |
| copy | O(n) |
| Get item | O(1) |
| Set item | O(1) |
| Delete item | O(1) |
| Contains (in) | O(1) |
| iteration | O(n) |





**Arrays:**

**Dynamic Arrays**

**Referential Arrays:**

Imagine 100 student names with ID numbers

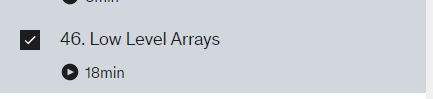
Each Cell of the array needs to have the same number of Bytes

How can we avoid having to have a series names?

We can use an array of Object References.



# 14/09/2022: Low Level Arrays



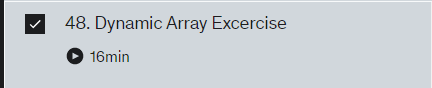
* Technical Overview
* Getting Help for the Technical Section

# 17/09/2022: Dynamic Array Exercise



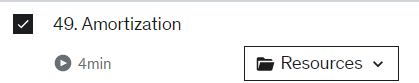
* Technical Overview
* Getting Help for the Technical Section

# 22/09/2022: Dynamic Array



* Technical Overview
* Getting Help for the Technical Section

# 22/09/2022: Amortization



* Technical Overview
* Getting Help for the Technical Section

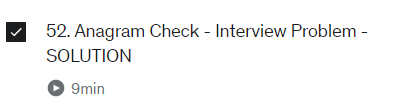
# 24/09/2022: Interview Problems



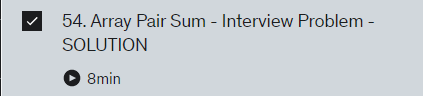
* Technical Overview
* Getting Help for the Technical Section

# 24/09/2022: Anagram Check

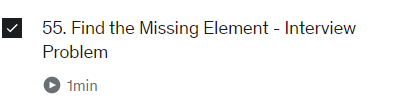




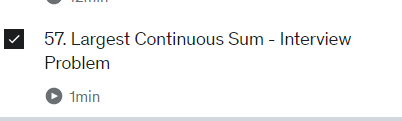
# 25/09/2022: Array Pair Sum

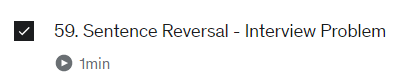


# 27/09/2022: Find the Missing Element

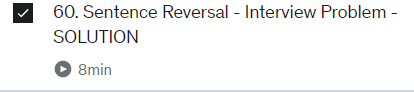


# 27/09/2022: Largest Continuous Sum and solution

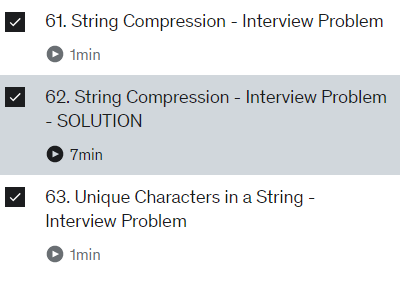


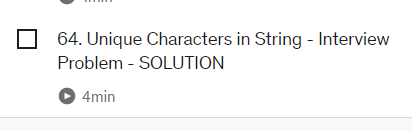


# 30/09/2022: Sentence Reversal



# 01/10/2022: String Compression and Unique Characters





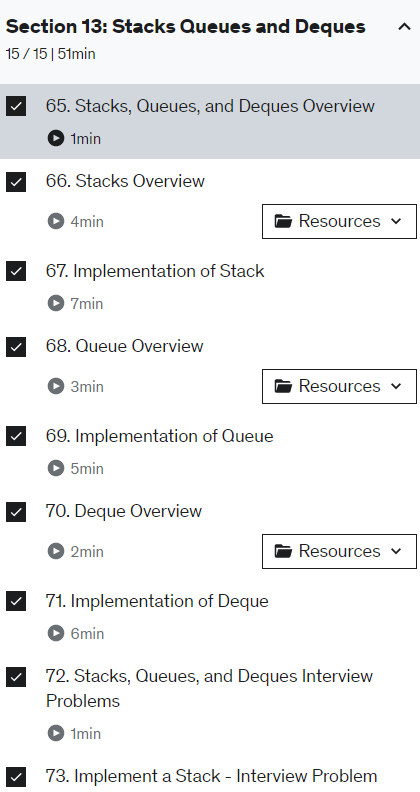
**05/10/2022 - 12/10/2022 : Stacks Queues and Dequeues**

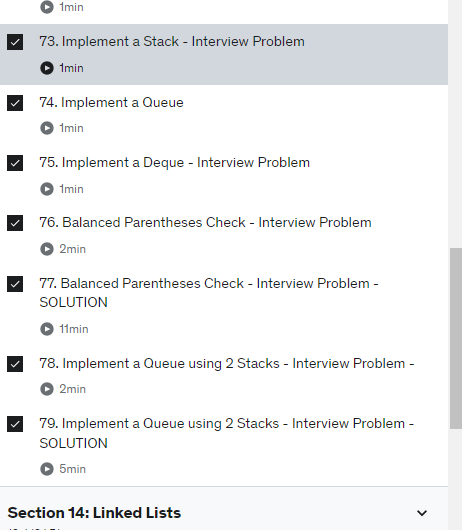
Stacks: They follow Last in First out Rule

Queues: It follow First in First out Rule

They can to be implemented with an array or linked list

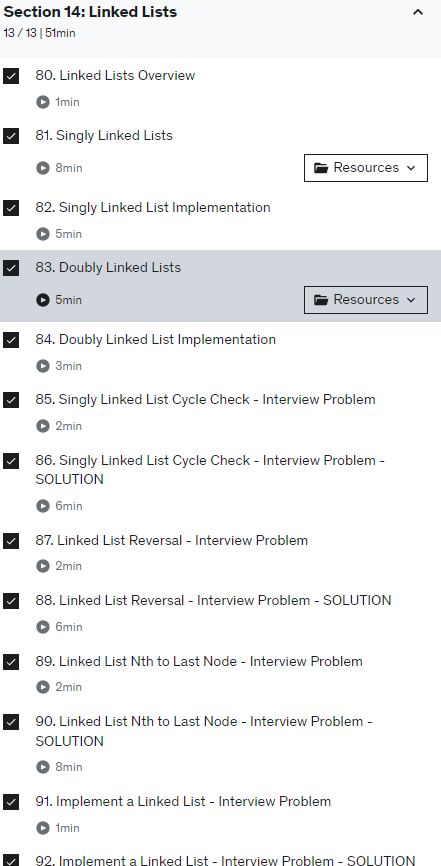
In this concept I have learned how to implement the Stacks, Queues using two stacks, Balanced paranthesis using queues





**14/10/2022 - 20/10/2022 Section :14 Linked Lists**

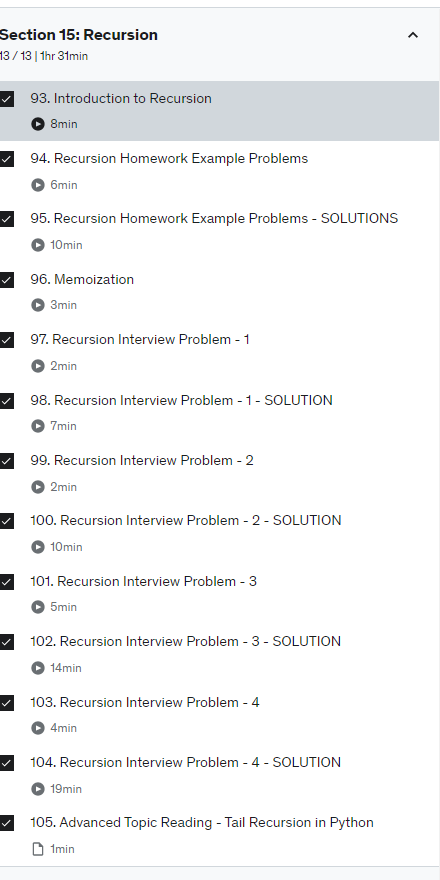
Linked Lists are a data structures that store data as a chain. The structure of a linked list is each value of data has a connection to the next one and previous as well which is known as Double linked list. The piece which points to only next node is called singly linked list. Each element in a linked list is called a node. I have practiced cycle check,



**21/10/2022 - 27/10/2022 Section 15: Recursion**

Recursion is process of defining something using the algorithm itself

In the recursion I have learned Factorial of Number, Cumulative sum, Recursive sum, Fibonacci, Coin Change Problem.



**28/10/2022 - 08/11/2022 Section 16 – Trees**

**Binary Tree-**

Binary trees have a special constraint that each node can have at most two children. Binary trees have the advantages of both ordered arrays and linked lists because searches are as fast as sorted arrays and insert or delete operations are as fast as linked lists.  
In this case, let's start with childless or childless. Let the left child start by saying equal to zero and the self-taught right child also equal to start. So the thing to note here is that just as a list can store any object, the constructor function expects to get some object to store in this root. An object in the tree can be a reference to any object.

I'm actually going to put in pass here for now and then we're also going to have inserts rights and it would also take a self new node and we'll put it pass there. So to add a left child to the tree we're going to create a new binary tree object and set the left attribute of the root to refer to this new object. So in order to do that we can start filling out insert left.

So I got to say if self stop left child is equal to none meaning we don't have anything there yet.

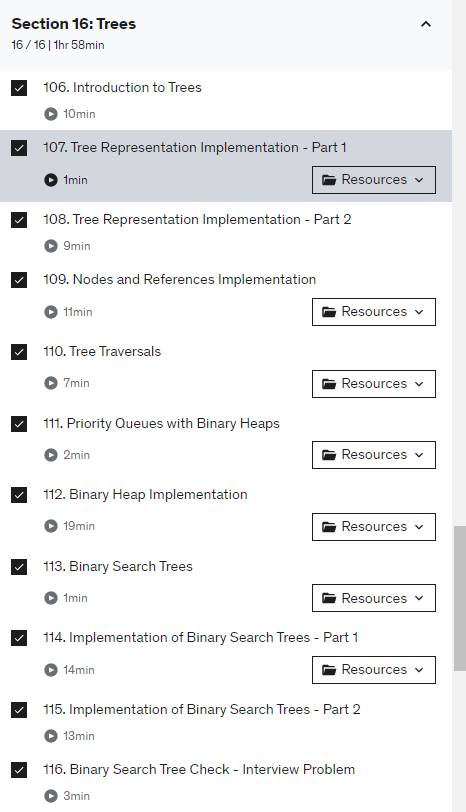
**Binary Search tree:**

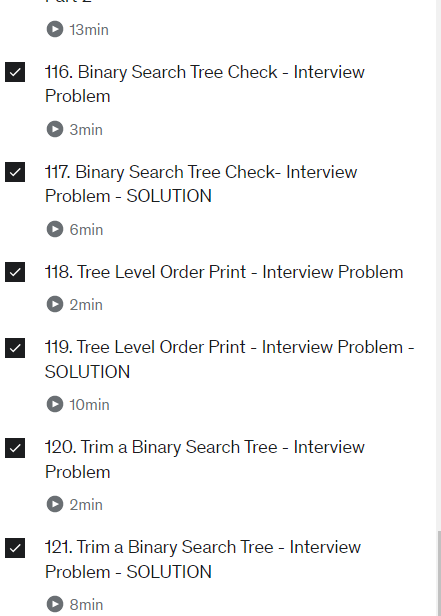
# binary search tree should always have the highest element on its right nodes and lowest on the left nodes

Deleting Node case 1 - the node to be deleted has no children

Case 2- The node o be deleted has only one child

case 3- The node to be deleted has two children





**Binary Search Trees**: To implement the binary search tree, we will use the nodes and the references approach similar to the one we used to implement the linked list , and the expression tree. Here We use two classes to implement the binary tree. The first class we will call Binary Search tree, and the second class we will call Tree Node.

**28/10/2022 - 08/11/2022 Section 17: Searching and Sorting**

In this module I have learned Sequential Search (Linear Search), hashing, Binary search Implementation of Binary Search, Quick sort, insertion sort, Bubble sort, Shell Sort, Merge sort, Selection sort.

**Linear Search**: The counter will iterate through entire list and searches for a target element.

**Binary Search**: The Algorithm will check whether the target is existed in first half or later half of the list and starts searching that part using the same process.

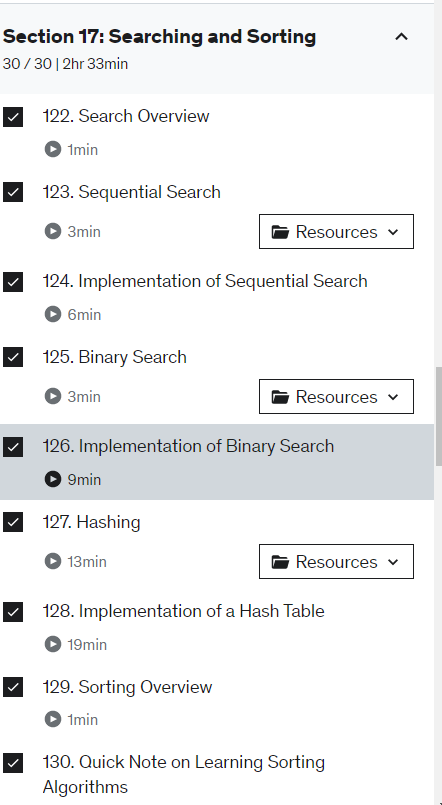
**Quick sort**: It is a divide and conquer algorithm

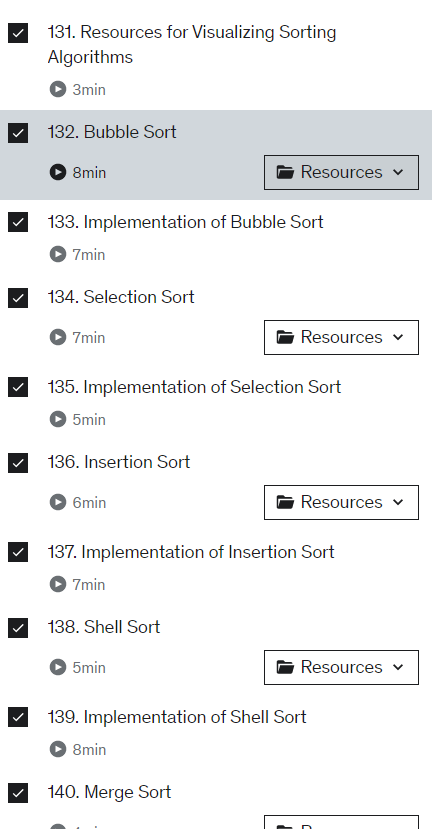
1. Always pick the first element as a pivot
2. Always pick the last element as a pivot
3. Pick a random element as a pivot
4. Pick median as a pivot

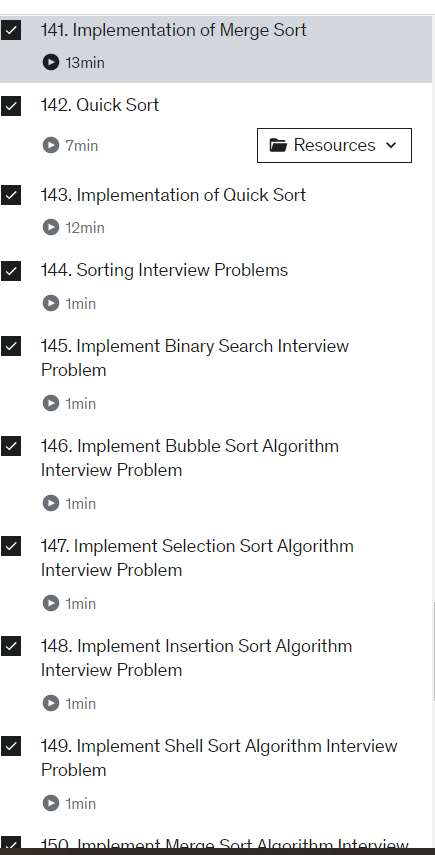
**Merge Sort**: The array is initially divided into two equal halves and then they are combined in a sorted manner.

**Bubble Sort** : It is the simplest sorting that works by repeatedly swapping the adjacent elements if they are in the wrong order. This algorithm is not suitable for large data sets as its average and worst-case time complexity is quite high.

**Shell sort**: It is particularly a variant of Insertion Sort. In insertion type, we flow factors simplest one role ahead. When an detail needs to be moved some distance ahead, many moves are involved. The concept of ShellSort is to permit the trade of some distance items. In Shell type, we make the array h-looked after for a massive fee of h. We hold lowering the fee of h till it will become 1. An array is stated to be h-looked after if all sublists of each h'th detail are looked after.

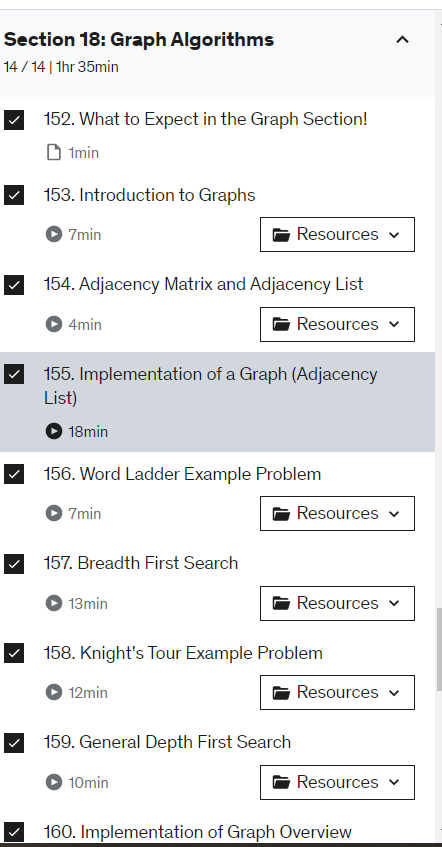


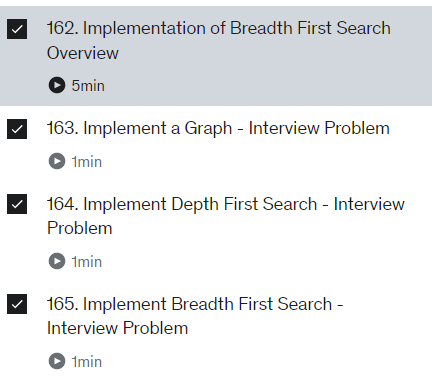




**09/11/2022 - 16/11/2022 Section 17 Section 18: Graphs**

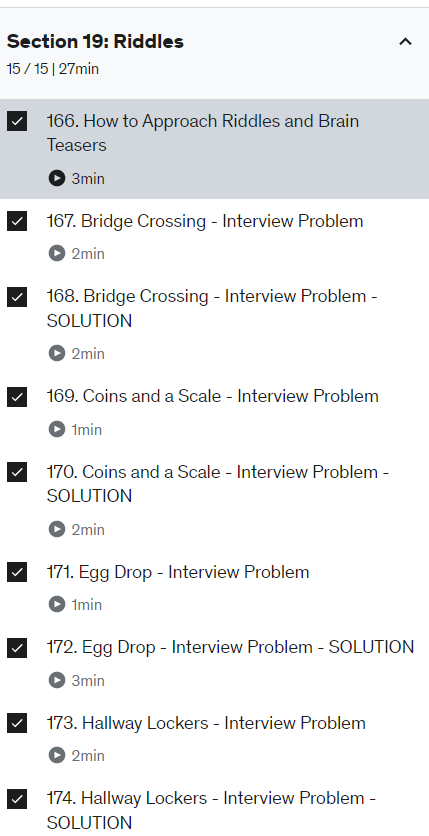
In this module I have learned concepts of Depth first search and Breadth First search , implementing adjacency list using graphs and the problems related to the concept.

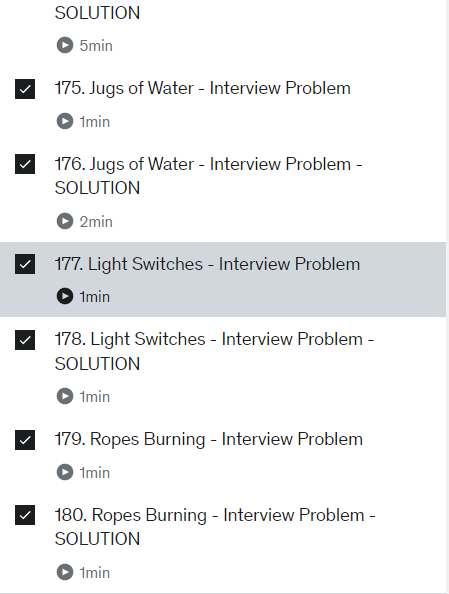




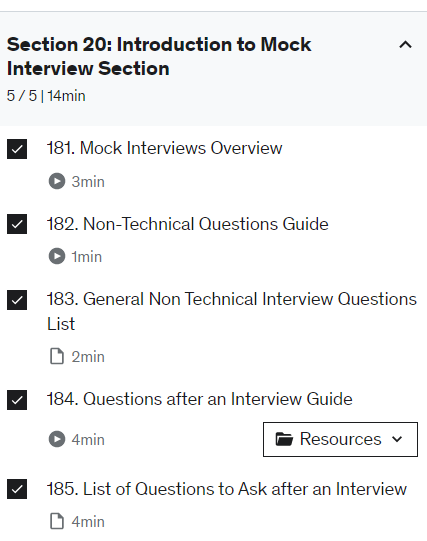
**17 /11/2022 – 24/11/2022 Section 19 Riddles:**

Riddles module helped me a lot to develop Aptitude skills and showcase in the interview. In this module I have solved Coin problem, Egg drop, Hallway Lockers , Bridge crossing, Jugs of water , Light switches, Ropes Burning. Every problem has given a different approach to solve the problem

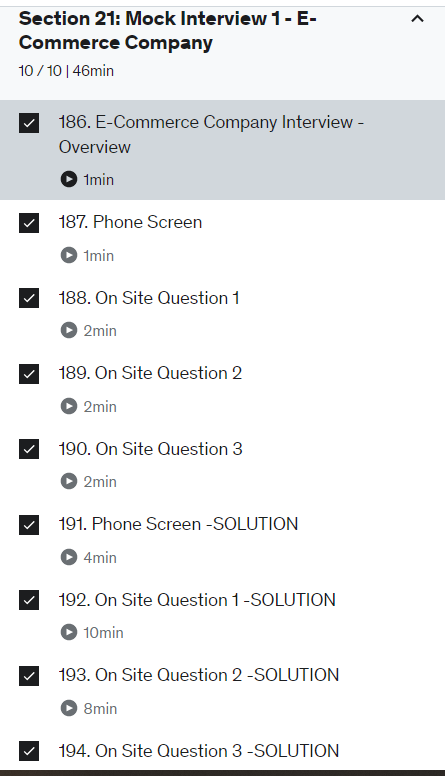




**25 /11/2022 – 30/11/2022 Section 20 Introduction to Mock Interview Section:**

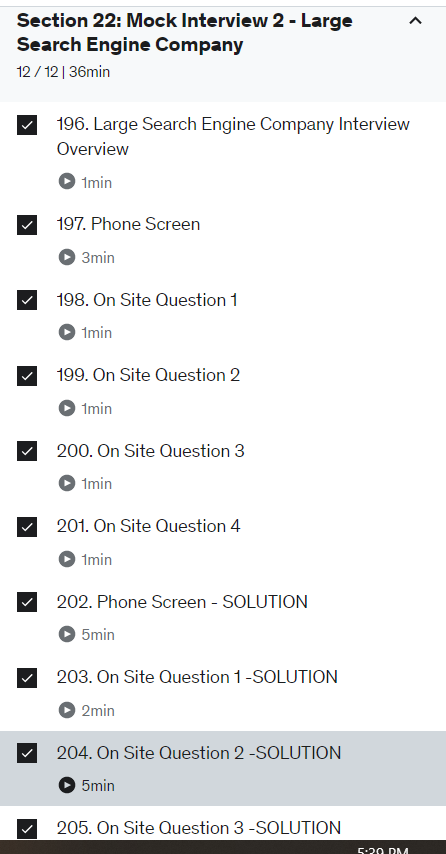


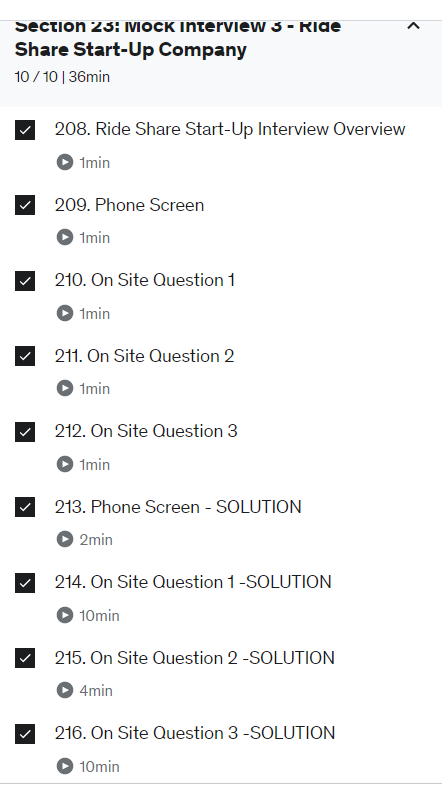
**25 /11/2022 – 30/11/2022 Section 21 E- Commerce Company:**

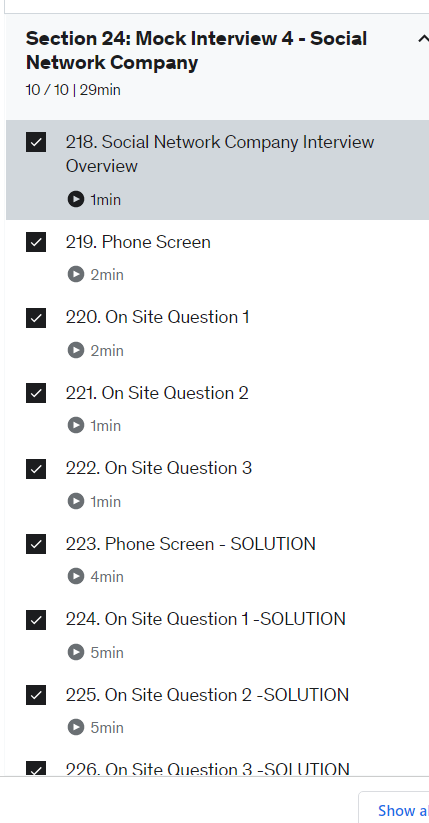


**30/11/2022 – 15/12/2022 Section 22 Introduction to Mock Interview Section:Large Search Engine Company, Ride share startup company, Social Network Company**

In these, section I have learned about few questions which test the aptitide of a candidate. I have written descriptions in programs.









The certificate is in Indian Timing because I have purchased with Indian Account