Usage of algorithms

05 February 2017 00:36

Real world examples of tree structures
1. PDF is a tree based format. It has a root node followed by a catalog node(these are often the same) followed by a pages node which has several child page nodes. Producers/consumers often use a balanced tree implementation to store a document in memory.

2. Computer chess games build a huge tree (training) which they prune at runtime using heuristics to reach an optimal move.
3. It goes without saying that connections/relations are very naturally modeled using graphs. Often, trees are used to represent/identify more interesting phenomena. How do you answer questions like "Does Harry and Sally have any common friend(s)?"

4. A common problem in bioinformatics is to search huge databases to find matches for a given query string. Tries are a common occurrence there.
5. Quite a few successful (stock) traders use decision trees in their day to day trading -- to choose a trade, to exit one. Often times these are not codified in a computer program, but written down somewhere on the back of their notebook.

Algorithms list

05 February 2017 00:40

Trees ,
Linked lists ,
DP ,
Arrays ,
Strings ,
Stacks ,
Queue ,
Hashing ,
Heaps , Backtracking
Graph questions

Behavioral

11 September 2017 22:06

Discussion about project
Project which has business impact
Conflict with manager/seniors
What you prefer to use for communication between android and backend server – XML/JSON ? why ?

Algorithm problems

04 June 2017 15:22

Algorithm sections	Number of problems	Problems done	Problems remained
Arrays	100	3	
Linked list	58		
Stack	30	4	
Binary tree	117		
Binary search tree	44		
Heaps	18		
Queue	20		
Graph	60		
Advanced DS	--//--		

Schedule

04 June 2017 15:30

Total days = 90
Total sections = 8
Each day Mandatory problem : 3

Binary Tree - 90	Stack - 30	LInkedLIst - 50	Binary Search Tree - 36
Queue - 20	Graph - 70	Array - 100	

DynamicProgramming- 150	Greedy- 42	Divide and Conquer - 53

Month	week	Problem list	Weekends	Every day	Every weekend * 4	
Feb	**All**	**Binary tree (30) array(30) + Linked list(30) Stack (10) || queue (7)**	**Graph = 10 Array = 10 Dynamic Programming - 40 Divide and conquer = 20 Greedy - 13 Revise**	**Binary tree (1)+ array(1) + Linked list(1) Stack (1) || queue (1)**	**Graph = 3 Array = 2 Dynamic Programming - 10 Divide and conquer =5 Greedy - 3**	
March	**All**	**Binary tree(30) + array (30)+ linked list (20)+ after(Binary search tree(10)) Stack (10) queue (7)**	**BT = 10 Graph = 10 Greedy - 13 Divide and conquer = 20 Dynamic Programming - 40 Revise**	**Binary tree(1) + array (1)+ linked list (1) -> after Linked List Binary search tree(1) Stack (1) || queue (1)**	**Graph = 3 Array = 2 Dynamic Programming - 10 Divide and conquer =5 Greedy - 3**	
April	**All**	**Binary tree(30) + array (30)+ Binary search tree(26) Stack (10) queue (7)**	**Graph = 10 Greedy - 13 Dynamic Programming - 40 Divide and conquer = 20 Greedy = 13 Revise**	**Binary tree(1) + array (1)+ Binary search tree(1) Stack (1) || queue (1)**	**Graph = 3 Array = 2 Dynamic Programming - 10 Divide and conquer =5 Greedy - 3**	

Tricky questions

10 September 2017 19:11

1. http://www.geeksforgeeks.org/given-only-a-pointer-to-a-node-to-be-deleted-in-a-singly-linked-list-how-do-you-delete-it/
   Fast solution is to copy the data from the next node to the node to be deleted and delete the next node. Something like following.

```
    // Find next node using next pointer
    struct Node *temp  = node_ptr->next;
// Copy data of next node to this node
    node_ptr->data  = temp->data;

// Unlink next node
    node_ptr->next  = temp->next;

// Delete next node
    free(temp);
```

2.

Solved problems

11 June 2017 18:10

<table>
<tr><th>Month</th><th>week</th><th>Problem list</th><th>Weekends</th><th>Done till the date</th><th>Problems remained</th></tr>
<tr><td>June</td><td>All</td><td>Binary tree (30)+ array(30) + Linked list(30)</td><td>Graph = 10
Array = 10
Search and sort = 10
Revise</td><td><table><tr><th>BT</th><th>Array</th><th>Linkedlist</th><th>Graph</th><th>SS</th></tr><tr><td>28</td><td>29</td><td>30</td><td>4</td><td>0</td></tr></table></td><td>http://www.geeksforgeeks.org/find-longest-path-directed-acyclic-graph/</td></tr>
<tr><td>July</td><td>All</td><td>Binary tree(30) + array (30)+ linked list (28)+
after(Binary search tree(2))</td><td>BT = 10
Graph = 10
Search and sort = 10
Revise</td><td></td><td></td></tr>
<tr><td>August</td><td>All</td><td>Binary tree(30) + array (30)+ Binary search tree(30)
Weekend</td><td>Graph = 10
Divide and conquer = 6
Greedy = 13
Revise</td><td></td><td></td></tr>
<tr><td>September</td><td>All</td><td>Binary search tree (12)+ queue(20)+ Heaps(18)+ Stack(30)</td><td>DP = 20
Graph = 20
Revise</td><td></td><td></td></tr>
<tr><td>October</td><td>All</td><td>Graph(30) + ADS(15) + Revision</td><td>Dp = 20
Backtracking = 10
Revise</td><td></td><td></td></tr>
<tr><td>November</td><td>2</td><td>Revise Everything</td><td></td><td></td><td></td></tr>
<tr><td>November</td><td>2</td><td>---------- Get ready for interviews!----------</td><td></td><td></td><td></td></tr>
</table>

Remained

09 July 2017 18:34

Which sort to use where ?
http://www.geeksforgeeks.org/knapsack-problem/

Given an unsorted array A and a number k, find the maximum sum sub string in the array such that its sum is divisible by k.

http://www.geeksforgeeks.org/maximum-sum-path-across-two-arrays/

http://www.geeksforgeeks.org/find-smallest-range-containing-elements-from-k-lists/

http://www.geeksforgeeks.org/fill-array-1s-minimum-iterations-filling-neighbors/

http://www.geeksforgeeks.org/find-median-bst-time-o1-space/

http://www.geeksforgeeks.org/print-possible-strings-can-made-placing-spaces/

Given two (binary) trees, return the first pair of non-matching leaves

http://www.geeksforgeeks.org/word-ladder-length-of-shortest-chain-to-reach-a-target-word/

http://www.geeksforgeeks.org/maximum-sum-nodes-binary-tree-no-two-adjacent/

http://www.geeksforgeeks.org/find-the-smallest-window-in-a-string-containing-all-characters-of-another-string/

http://www.geeksforgeeks.org/find-the-largest-subtree-in-a-tree-that-is-also-a-bst/

Operating System questions

03 September 2017 20:22

1	What is critical section problem and how is it solved?
2	What is Semaphore?
3	What is race condition?
4	What is write lock and what are the conditions of it?
5	What are threads and processes?
6	Create queue using stack.
7	Discussion on all the OS scheduling Algorithms.
8	How to implement SRTFS (preemptive SJF).
9	Discussion on Inter Process Communication : Shared Memory and Socket
10	Why sockets are preferred over Shared Memory.
11	Where shared memory is stored (in user space or kernel space) ? and why ?
	process & threads, Inter process communication, Lots of discussion on scheduling algorithm, their implementations, data structures used to implement scheduling algorithms, scheduling algorithms in real time distributed systems & implementation.
12	virtual memory, paging page faults
13	Design Memory Management System and tell about all the data structures you will use and why. How will you allocate and deallocate the memory using those data structure and Time & Space complexity of the operations.
14	4. Threads vs Process. They were impressed because I explained with a real-time example 5. How the Program is stored in memory? Stack frame for Program? 6. When two threads access a code at the same time, what happens. How is it prevented? 7. Difference between mutex and semaphore. 8. Priority Inheritance and Priority Inversion. I have a DB of some size as of now. In future new DB with additional space will be added. How will the DB be modified? What to modify? How to scale it? 4. Distributed computing concepts 5. Multiple partitions in OS 6. MVC design pattern

Quick tips

07 September 2017 16:37

Problem type	DS	Can be used
Range sum problems	segment tree	
Kth smallest / greatest -	heap	
Autocomplete or searching for relevant words in list	TRIE	

DS and Algo Tips

10 September 2017 19:53

Problem statement	Algorithm	Reference
Counting number of connected components in a undirected weighted graph	DFS	http://www.geeksforgeeks.org/find-number-of-islands/
Finding kth smallest/ highest	Heap	Can be done with BinarySearchTree
Copy LL with random and next	Add previous copy node after every node	
LRU cache	DLL as queue and LRU page is at end and MRU page is at front. HashMap for storing reference to node and reference to any node	
Questions involving finding a sequence e.g a->b B -> c C -> d Whos first ? A -> B -> C- > D	Topological sorting for arranging or identifying the sequence	
Finding middle of LL/ identifying loop in LL	Fast and slow pointer	
Finding sum of ranges in array or string	Segment tree	
Auto correct or suggestions	Trie	
Given an range of number count the numbers which has same first and last digits.	Every span of 10 has one number	
LCA in BST	Are n1 and n2 smaller than root ? Move to left if both greater than root ? Move right If one is less than root and other is greater then we found LCA!	http://www.geeksforgeeks.org/lowest-common-ancestor-in-a-binary-search-tree/
Find number in array where all element < number and all element after number are >	Create two arrays leftmax and rightmin Where leftmax[i] contains max value from 0 to I Where rightMin[i] contains min value from i+1 to n	
Reverse the given string	1) Reverse the individual words, we get the below string. "i ekil siht margorp yrev hcum" 2) Reverse the whole string from start to end and you get the desired output. "much very program this like i"	
Add 1 to linkedlist without recursion without extra space	1. Reverse the linkedlist 2. From head of reversed linkedlist add 1 and forward carry 3. Reverse linkedlist	http://www.geeksforgeeks.org/add-1-number-represented-linked-list/

Rotate matrix by 90 degree	Int ring = n /2; For each ring rotate one by one element	http://www.geeksforgeeks.org/inplace-rotate-square-matrix-by-90-degrees/
Find k most frequent words in a stream	The word is already present. We just increase the corresponding frequency value in min heap and call minHeapify() for the index obtained by "indexMinHeap" field in Trie. When the min heap nodes are being swapped, we change the corresponding minHeapIndex in the Trie. Remember each node of the min heap is also having pointer to Trie leaf node. **2.** The minHeap is not full. we will insert the new word into min heap & update the root node in the min heap node & min heap index in Trie leaf node. Now, call buildMinHeap(). **3.** The min heap is full. Two sub-cases arise. ….**3.1** The frequency of the new word inserted is less than the frequency of the word stored in the head of min heap. Do nothing. ….**3.2** The frequency of the new word inserted is greater than the frequency of the word stored in the head of min heap. Replace & update the fields. Make sure to update the corresponding min heap index of the "word to be replaced" in Trie with -1 as the word is no longer in min heap. **4.** Finally, Min Heap will have the k most frequent words of all words present in given file. So we just need to print all words present in Min Heap.	http://www.geeksforgeeks.org/find-the-k-most-frequent-words-from-a-file
Next greater element		http://www.geeksforgeeks.org/next-greater-element/
Why BFS in unweighted and why DFS in weighted ?	• The basic idea is similar to the unweighted case • A major difference is this: ▪ In an unweighted graph, breadth-first search guarantees that when we first make it to a node *v* , we can be sure we have found the shortest path to it; more searching will never find a path to *v* with fewer edges ▪ In a weighted graph, when we first make it to a node *v* , we can't be sure we have found the best path to *v* : there could be a path with more edges, but less overall cost, that we would find later • Still, Dijkstra's is a greedy algorithm: ▪ At each stage of the algorithm, we will extend the *best* path we have found so far: this guarantees we will know when we have found the shortest (least-cost) path from the source vertex, if there are no negative-cost edges ▪ Keeping track of paths in terms of which is best will require a priority queue -- a very common data structure in greedy algorithms • (Note that Djikstra's algorithm will work for an unweighted graph: treat it as a weighted graph with all edge weights the same, e.g. 1) From <http://cseweb.ucsd.edu/~kube/cls/100/Lectures/lec12/lec12-28.html>	
Find smallest window which contains all characters from other string	(solved it - missed edge cases) Find currentMinWindow and store count of current window in hashmap also for pattern. One window is found check if we can remove repeated charactes by looking at currWindow hashmap and junk characters and move	1. http://www.geeksforgeeks.org/find-the-smallest-window-in-a-string-

	the start towards right.	containing-all-characters-of-another-string/ From <http://www.geeksforgeeks.org/amazon-interview-experience-set-377-campus-full-time/>
Given a sorted dictionary of an alien language, find order of characters From		<http://www.geeksforgeeks.org/given-sorted-dictionary-find-precedence-characters/>
Diameter of binary tree	The diameter of a tree T is the largest of the following quantities: * the diameter of T's left subtree * the diameter of T's right subtree * the longest path between leaves that goes through the root of T (this can be computed from the heights of the subtrees of T)	http://www.geeksforgeeks.org/diameter-of-a-binary-tree/
Merge k sorted arrays	Creaete a heap of k elements with index of array from which elemnt is takes heapify and for all 1..n elements in each array removed min put it in array and pluck element from index of removed min replace min with that element minheapify again. Viola!	
Find duplicate element in an array where range is 1 to n	Traverse array make every visited	
	1 . how to handle server scaling up to sudden increased of request 2 . how to handle scaling the data 3. Design BookMyShow tables .	
Given an array of strings, find if the given strings can be chained to form a circle. A string X can be put before another string Y in circle if the last character of X is same as first character of Y.	1) Create a directed graph g with number of vertices equal to the size of alphabet. We have created a graph with 26 vertices in the below program. 2) Do following for every string in the given array of strings. …..a) Add an edge from first character to last character of the given graph. 3) If the created graph has eulerian circuit, then return true, else return false.	From <http://www.geeksforgeeks.org/given-array-strings-find-strings-can-chained-form-circle/>
Find distance between two given keys of a Binary Tree	**Dist(n1, n2) = Dist(root, n1) + Dist(root, n2) - 2*Dist(root, lca)** 'n1' and 'n2' are the two given keys 'root' is root of given Binary Tree. 'lca' is lowest common ancestor of n1 and n2 Dist(n1, n2) is the distance between n1 and n2.	From <http://www.geeksforgeeks.org/find-distance-two-given-nodes/>
Infix to postfix	**1.** Scan the infix expression from left to right. **2.** If the scanned character is an operand, output it.	http://www.geeksforgeeks.org/stack-set-2-infix-to-postfix/

	3. Else, …..**3.1** If the precedence of the scanned operator is greater than the precedence of the operator in the stack(or the stack is empty), push it. …..**3.2** Else, Pop the operator from the stack until the precedence of the scanned operator is less-equal to the precedence of the operator residing on the top of the stack. Push the scanned operator to the stack. **4.** If the scanned character is an ‘(‘, push it to the stack. **5.** If the scanned character is an ‘)’, pop and output from the stack until an ‘(‘ is encountered. **6.** Repeat steps 2-6 until infix expression is scanned. **7.** Pop and output from the stack until it is not empty.	
Data compression techniques	Applications ?	https://www.ijitr.com/index.php/ojs/article/viewFile/486/pdf
K Most frequent words	Trie and minheap	
Find element in row and col wise sorted matrix	Start from top right . If arr element > x move left else down	http://www.geeksforgeeks.org/amazon-interview-experience-set-390-campus/

Links

11 September 2017 19:31

https://www.hiredintech.com/classrooms/system-design/lesson/53
https://github.com/checkcheckzz/system-design-interview
http://massivetechinterview.blogspot.co.uk/2015/07/design-chess-game-using-oo-principles.html -- search for problems here

https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-827-multithreaded-parallelism-languages-and-compilers-fall-2002/ - not related
Applications of min spanning tree - http://www.geeksforgeeks.org/?p=11110

Problems

11 September 2017 20:01

Amazon design question list: https://www.careercup.com/page?pid=amazon-interview-questions

Scalability: http://www.lecloud.net/post/7295452622/scalability-for-dummies-part-1-clones

https://medium.com/vimeo-engineering-blog/improving-load-balancing-with-a-new-consistent-hashing-algorithm-9f1bd75709ed

http://www.aosabook.org/en/distsys.html

Problem statement	Useful Links
Design Elevator with OOPS concept	1. http://www.cs.cmu.edu/~luluo/Courses/18540PhDreport.pdf 2. http://www.angelfire.com/trek/software/elevator.html 3. http://practice.geeksforgeeks.org/problems/design-elevator
Design Bookmyshow website	1. http://practice.geeksforgeeks.org/problems/design-bookmyshow 2. https://www.quora.com/How-can-I-design-on-BookMyShow 3. https://www.careercup.com/question?id=12321687
Design a musical juke box with functions	1. https://www.careercup.com/question?id=5705431074734080 2. https://tianrunhe.wordpress.com/2012/03/19/design-a-musical-jukebox-using-oo-principles/ 3. http://massivetechinterview.blogspot.co.uk/2015/07/design-musical-jukebox-using-oo.html
Design Outlook meeting request handler	1. https://www.careercup.com/question?id=15555745
Design chess	1. https://codereview.stackexchange.com/questions/71790/design-a-chess-game-using-object-oriented-principles 2. http://massivetechinterview.blogspot.co.uk/2015/07/design-chess-game-using-oo-principles.html 3. https://github.com/tim-hr/stuff/wiki/System-design:-How-would-you-design-a-two-player-online-chess-game%3F
Design Data warehouse placement system.	????
Design comment system.	I have already done it in

	connectin
Design twitter	https://www.interviewbit.com/problems/design-twitter/ http://blog.gainlo.co/index.php/2016/02/17/system-design-interview-question-how-to-design-twitter-part-1/ http://blog.gainlo.co/index.php/2016/02/24/system-design-interview-question-how-to-design-twitter-part-2/ https://hackernoon.com/anatomy-of-a-system-design-interview-4cb57d75a53f
Design news feed system	http://blog.gainlo.co/index.php/2016/03/29/design-news-feed-system-part-1-system-design-interview-questions/
Doctor appointment system	
design Truecaller kind of system	
Performance management system (appraisal workflow system)	
Design an in-memory file system	https://tianrunhe.wordpress.com/2012/03/24/design-an-in-memory-file-system/
Design Uber	http://massivetechinterview.blogspot.co.uk/2015/09/the-uber-software-architecture.html https://www.careercup.com/question?id=6211118064205824
1. Design Conference Room Allocation system considering the time and number of rooms available constraints.	
1. Design Snake And Ladder Game.	
Design producer/consumer live and offline	
Design comment system.	
video streaming system.	
Build an **UBER** platform no code discussion about how will you do it, 1. How will handle locations (most updated) of the driver ? 2. How will update the location of the driver (architecture of app or service to update) ? 3. Find nearest driver ? (We can filter all drivers based on location) 4. What if two cities are merged like Redmond and Bellevue then other car might be close but in second city where city filter want work ? 5. How do you scale same app to other countries (like china) ? 6. One you find the nearest driver how do pick them ? (I said ratings) 7. What would do consider as good ratings ? 8. Once you have 10 driver with 4.8 ratings how will you send the request ?	

9. What will you do if all 10 drivers receives and accepts 10 request simultaneously ?	
Do integration for Splitwise app with Amaonz Pay (or Paytm) 1) where a person can pay to another person and money directly gets deposited into other person's bank account. 2) A person can also send reminder to another person for owning money.	
Design a job scheduler, scalability, fault tolerance, high availability, how scheduler picks up job, how will you take care where one job can run for 30 min and one for 30 hour, how will you distribute jobs on servers. Based on frequency & time how will you execute them ? How will you notify back the user about start/stop or completion of a job ? How will your system know if a job is killed / terminated due to unknown reasons ?	
1. Design VLC.	
Design Restaurant booking application HLD and LLD. Write an algorithm to optimally assign tables for bookings.	
Design a ranking system. We have an infinite supply of words ending with '.' We need to implement a reader program that ranks words on the basis of certain criteria • Example: This is my cat. • This house belongs to my uncle • An amazing country with so many tourist places And so on.. • Ranking System criteria : rank the words on the basis of occurrence, for example Output : This:2, is:2, my:2… highest rank (sorted asc or desc based on provided flag) Design it completely and scalable Ranking System.	
• Design a debugger	
• Design Amazon Cart System. (HLD and LLD) Discussion on solution.	
Design Recommendation system. How will you generate generate recommendations for millions of users. DB Schema, How will you improve latency? if the user is searching a item, when will you show next recommendations. How will you update, latency basically and consistency.	
Design the payment module of the Uber App.	
Design Memory Management System and tell about all the data structures you will use and why. How will you allocate and deallocate the memory using those data structure and Time & Space complexity of the operations	

1. Design a backend store system for managing inventory and orders with different pricing structure of items across stores. Focus was on schema and API design along with which parts of the system will be centralised and which will be at store level

2. Design producer/consumer live and offline video streaming system.
   Asked me to design the flight system with src, destination and no of hops
3. Design Dating application. HLD, then specific discussion of profile search and ranking based on the users' interest. Discussion on making it scalable.
4. Design a Railway system
5. Design survey monkey like website.More emphasis was on Admin related activities of creating/editing a survey, showing all the surveys created by the admin user.
   Generate a unique link to each of the survey created by the admin.
   Started from UI pages designing to what information should be sent through Ajax calls to server for each of CRUD operations on survey.
   Structure of each table to store the survey information. How will I as a developer create a layered architecture at server side.
   Layered architecture: Ajax -> Servlet -> Class for CRUD operations on surveys –> Transaction Handler –> Persistence laye

6. Design a Fresh Grocery System. Means Daily usable Items, you cannot store them in inventory like bread, milk etc.
   HLD+ DB Schema + Concurrency issues + Scalable architecture. How will you scale to multiple countries
7. Design an Online tracking system, similar to how you can track your order on swiggy after you place it on maps.

8. Design an online file repository system
   - What are the use cases
   - Features
   - Security
   - HLD
   - LLD
9. Design cricket info live score page
10. Design Cache library
    Design Debugger

Problems to revise

12 September 2017

http://www.geeksforgeeks.org/minimum-number-of-swaps-required-for-arranging-pairs-adjacent-to-each-other/ --- Backtracking (not done)

http://www.geeksforgeeks.org/dynamic-programming-set-31-optimal-strategy-for-a-game/ -- not done

http://www.geeksforgeeks.org/print-nodes-distance-k-given-node-binary-tree/

http://www.geeksforgeeks.org/counting-inversions/

http://www.geeksforgeeks.org/count-possible-decodings-given-digit-sequence/

http://www.geeksforgeeks.org/dynamic-programming-set-27-max-sum-rectangle-in-a-2d-matrix/

http://www.geeksforgeeks.org/detect-and-remove-loop-in-a-linked-list/

Done from problems to revise

21 September 2017 16:18

http://www.geeksforgeeks.org/gold-mine-problem/ -- solved it no need to traverse from back solved it using front :)

19 September 2017 09:39

https://www.allaboutcircuits.com/textbook/digital/chpt-2/negative-binary-numbers/

Standard algo complexity

20 September 2017 10:15

```
Trie - insert search O(key_len)
Space complexity - O(AlphabetSize * key_len * N)
```

Common Data Structure Operations

Data Structure	Time Complexity								Space Complexity
	Average				Worst				Worst
	Access	Search	Insertion	Deletion	Access	Search	Insertion	Deletion	
Array	Θ(1)	Θ(n)	Θ(n)	Θ(n)	O(1)	O(n)	O(n)	O(n)	O(n)
Stack	Θ(n)	Θ(n)	Θ(1)	Θ(1)	O(n)	O(n)	O(1)	O(1)	O(n)
Queue	Θ(n)	Θ(n)	Θ(1)	Θ(1)	O(n)	O(n)	O(1)	O(1)	O(n)
Singly-Linked List	Θ(n)	Θ(n)	Θ(1)	Θ(1)	O(n)	O(n)	O(1)	O(1)	O(n)
Doubly-Linked List	Θ(n)	Θ(n)	Θ(1)	Θ(1)	O(n)	O(n)	O(1)	O(1)	O(n)
Skip List	Θ(log(n))	Θ(log(n))	Θ(log(n))	Θ(log(n))	O(n)	O(n)	O(n)	O(n)	O(n log(n))
Hash Table	N/A	Θ(1)	Θ(1)	Θ(1)	N/A	O(n)	O(n)	O(n)	O(n)
Binary Search Tree	Θ(log(n))	Θ(log(n))	Θ(log(n))	Θ(log(n))	O(n)	O(n)	O(n)	O(n)	O(n)
Cartesian Tree	N/A	Θ(log(n))	Θ(log(n))	Θ(log(n))	N/A	O(n)	O(n)	O(n)	O(n)
B-Tree	Θ(log(n))	Θ(log(n))	Θ(log(n))	Θ(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)
Red-Black Tree	Θ(log(n))	Θ(log(n))	Θ(log(n))	Θ(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)
Splay Tree	N/A	Θ(log(n))	Θ(log(n))	Θ(log(n))	N/A	O(log(n))	O(log(n))	O(log(n))	O(n)
AVL Tree	Θ(log(n))	Θ(log(n))	Θ(log(n))	Θ(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)
KD Tree	Θ(log(n))	Θ(log(n))	Θ(log(n))	Θ(log(n))	O(n)	O(n)	O(n)	O(n)	O(n)

Sorting complexities

20 September 2017 10:16

Array Sorting Algorithms

Algorithm	Time Complexity			Space Complexity
	Best	Average	Worst	Worst
Quicksort	Ω(n log(n))	Θ(n log(n))	O(n^2)	O(log(n))
Mergesort	Ω(n log(n))	Θ(n log(n))	O(n log(n))	O(n)
Timsort	Ω(n)	Θ(n log(n))	O(n log(n))	O(n)
Heapsort	Ω(n log(n))	Θ(n log(n))	O(n log(n))	O(1)
Bubble Sort	Ω(n)	Θ(n^2)	O(n^2)	O(1)
Insertion Sort	Ω(n)	Θ(n^2)	O(n^2)	O(1)
Selection Sort	Ω(n^2)	Θ(n^2)	O(n^2)	O(1)
Tree Sort	Ω(n log(n))	Θ(n log(n))	O(n^2)	O(n)
Shell Sort	Ω(n log(n))	Θ(n(log(n))^2)	O(n(log(n))^2)	O(1)
Bucket Sort	Ω(n+k)	Θ(n+k)	O(n^2)	O(n)
Radix Sort	Ω(nk)	Θ(nk)	O(nk)	O(n+k)
Counting Sort	Ω(n+k)	Θ(n+k)	O(n+k)	O(k)
Cubesort	Ω(n)	Θ(n log(n))	O(n log(n))	O(n)

Others

20 September 2017 10:17

Problems	Time complexity O()	Space complexity			
Build a heap	O(n)	O(n)			
heapify	O(log (n))	O(n)			

Where to use which sort

22 September 2017 11:18

Quick sort: When you don't need a stable sort and average case performance matters more than worst case performance. A quick sort is O(N log N) on average, O(N^2) in the worst case. A good implementation uses O(log N) auxiliary storage in the form of stack space for recursion.
Merge sort: When you need a stable, O(N log N) sort, this is about your only option. The only downsides to it are that it uses O(N) auxiliary space and has a slightly larger constant than a quick sort. There are some in-place merge sorts, but AFAIK they are all either not stable or worse than O(N log N). Even the O(N log N) in place sorts have so much larger a constant than the plain old merge sort that they're more theoretical curiosities than useful algorithms.
Heap sort: When you don't need a stable sort and you care more about worst case performance than average case performance. It's guaranteed to be O(N log N), and uses O(1) auxiliary space, meaning that you won't unexpectedly run out of heap or stack space on very large inputs.
Introsort: This is a quick sort that switches to a heap sort after a certain recursion depth to get around quick sort's O(N^2) worst case. It's almost always better than a plain old quick sort, since you get the average case of a quick sort, with guaranteed O(N log N) performance. Probably the only reason to use a heap sort instead of this is in severely memory constrained systems where O(log N) stack space is practically significant.
Insertion sort: When N is guaranteed to be small, including as the base case of a quick sort or merge sort. While this is O(N^2), it has a very small constant and is a stable sort.
Bubble sort, selection sort: When you're doing something quick and dirty and for some reason you can't just use the standard library's sorting algorithm. The only advantage these have over insertion sort is being slightly easier to implement.

Non-comparison sorts: Under some fairly limited conditions it's possible to break the O(N log N) barrier and sort in O(N). Here are some cases where that's worth a try:
Counting sort: When you are sorting integers with a limited range.
Radix sort: When log(N) is significantly larger than K, where K is the number of radix digits.
Bucket sort: When you can guarantee that your input is approximately uniformly distributed.
From <https://stackoverflow.com/questions/1933759/when-is-each-sorting-algorithm-used>

Bubble sort

12 March 2018 12:09

Bubble sort
Worst and Average Case Time Complexity: O(n*n). Worst case occurs when array is reverse sorted.
Best Case Time Complexity: O(n). Best case occurs when array is already sorted.
Auxiliary Space: O(1)
Boundary Cases: Bubble sort takes minimum time (Order of n) when elements are already sorted.
Sorting In Place: Yes
Stable: Yes
From <https://www.geeksforgeeks.org/bubble-sort/>

Due to its simplicity, bubble sort is often used to introduce the concept of a sorting algorithm.
In computer graphics it is popular for its capability to detect a very small error (like swap of just two elements) in almost-sorted arrays and fix it with just linear complexity (2n). For example, it is used in a polygon filling algorithm, where bounding lines are sorted by their x coordinate at a specific scan line (a line parallel to x axis) and with incrementing y their order changes (two elements are swapped) only at intersections of two lines.

Selection Sort

12 March 2018 12:13

The selection sort algorithm sorts an array by repeatedly finding the minimum element (considering ascending order) from unsorted part and putting it at the beginning. The algorithm maintains two subarrays in a given array.

Time Complexity: $O(n^2)$ as there are two nested loops.
Auxiliary Space: O(1)
The good thing about selection sort is it never makes more than O(n) swaps and can be useful when memory write is a costly operation.

Insertion sort

12 March 2018 12:16

12, 11, 13, 5, 6
Let us loop for i = 1 (second element of the array) to 5 (Size of input array)
i = 1. Since 11 is smaller than 12, move 12 and insert 11 before 12
11, 12, 13, 5, 6
i = 2. 13 will remain at its position as all elements in A[0..I-1] are smaller than 13
11, 12, 13, 5, 6
i = 3. 5 will move to the beginning and all other elements from 11 to 13 will move one position ahead of their current position.
5, 11, 12, 13, 6
i = 4. 6 will move to position after 5, and elements from 11 to 13 will move one position ahead of their current position.
5, 6, 11, 12, 13

From <https://www.geeksforgeeks.org/insertion-sort/>

Time Complexity: O(n*n)
Auxiliary Space: O(1)
Boundary Cases: Insertion sort takes maximum time to sort if elements are sorted in reverse order. And it takes minimum time (Order of n) when elements are already sorted.
Algorithmic Paradigm: Incremental Approach
Sorting In Place: Yes
Stable: Yes
Online: Yes
Uses: Insertion sort is used when number of elements is small. It can also be useful when input array is almost sorted, only few elements are misplaced in complete big array.
What is Binary Insertion Sort?
We can use binary search to reduce the number of comparisons in normal insertion sort. Binary Insertion Sort find use binary search to find the proper location to insert the selected item at each iteration. In normal insertion, sort it takes O(i) (at ith iteration) in worst case. we can reduce it to O(logi) by using binary search. The algorithm as a whole still has a running worst case running time of O(n2) because of the series of swaps required for each insertion. Refer this for implementation.

Behavioral

20 September 2017 17:17

https://www.thestudentroom.co.uk/showthread.php?t=855623

If you are behind the schedule on a project, what will you do?	What is the cause of behind schedule? Wrong estimate - Increasing the resources to fill in the gap External dependency - Was this risk identified already. Communicate now Less resources: Engage more resources something else? Is it okay to deliver little late, or time is critical?
When you were in a conflict with a teammate or team member and he persuaded you to go his way? What would you do?	Don't take anything personally. If there is any argument on anything then try to explain the advantages/disadvantages by taking an example with everyone's approach. If both are suggesting the different approaches and the result is going to be the same without any performance impact then follow other's approach only, So that they will get the positive impression on you and it will be useful for the next time. Involve third person for the opinion.
what will you do when client is asking you to provide something which doesnt make any sense?	Explain user that the change requested by him/her doesn't improve application performance or functionality and suggest alternatives to the problem which user referred to. Convince user that we would take his/her change after consulting with all stakeholder of the application.
Describe a time when you saw an issue and fixed it	Last month in a quarterly release at the end moment we encountered an issue related to data (sorting when values are null) I quiskly analyzed an issue and found out the root cause . After I found the fix I discussed with concrned stakeholders about the fix when they said go ahead and went and fixed issue. After fixing I drafted out an email related to how we can avoid such things at the critical time may be by predefining some test case s, by automation. may be adding a null check in standard definition ?
What is your work style ?	I always try to be creative in work I do. Being thrown in highly competitive environment I have found my ways for dealing with working under pressure and still deliver in good way under deadline.
How to make sure the work done by you in a project is easily understandable by a person after you leave the project?	Documentation. KT session.
Tell me when you took risk	Decided to change the predefined architecture of the system when worked with mediamath team of bring your own data. Represented the demo to client. It went well and client liked it.
Tell me when you inspired your team	Had taken the training sessions for particular technology that I know of . Also conducted sessions for the same explaining basic ideas in an understandable manner and I was appreciated by team.
How do you handle a low performing team member ?	Have a one to one meeting with him. Check with him if he is facing any roadblocks in completing his tasks. If required arrange a training course for him or assign a mentor to work with him. Once training is over, give him ownership of some tasks and see if there is any improvement. If he expresses interest in some other area of work, help him in getting in touch with managers in that area and ensure a smooth transition for him and for the team.
If you are behind the schedule on a project, what will you do?	What is the cause of behind schedule? Wrong estimate - Increasing the resources to fill in the gap External dependency - Was this risk identified already. Communicate now Less resources: Engage more resources something else? Is it okay to deliver little late, or time is critical? Use agile, so that the delay is not under your responsibility instead it will be resolved in scrum of scrums level - Check grunt chart and adjust the critical path possibly - Push non critical items to the next release - Try to borrow some additional resources from peer team with portable skills

	If the process is flexible try to adjust the schedule internally i.e. optimise the time by cutting the turn around times with testing team, database approval team, devops reviews etc - Make the people to come on Saturday and stay late, usually Indians love this however you can't repeat this often - Once accomplished report the incident and archive with good details of root cause analysis. Usually the cause will be poor estimation, external dependencies, process flaws distraction & interruption.
Tell Me About a Time You Had to Lead and Motivate People.	**Situation** - When I was in finoux we had to deliver the netbanking project and simultaneously work on other projects as well. It was difficult to divide attention between two major things . And team started fealing a bit hazy about this. **Approach** - Then I could come up with an action where one resource will have a pair working with her on netbanking and when other project requirement comes up we will engage that resource. **Result** -team was very much positive about his approach. it turned around to be a useful suggestion where we were able to deliver the netbanking project as well as follow requirements of other client as well.
Tell Me About a Time You Had a Conflict on a Team Project.	I was managing the creation of our new corporate brochure and we were on a very tight deadline because we had to have brochures printed in time for a big upcoming trade show. • I was in charge of delivering on time and I had to manage team members from Marketing, Sales, Graphic Design, and Product Management. • The designer that was assigned to the project was very talented, but unfortunately missed a deadline that I assigned. When I approached him about it, he blew up at me. I was taken aback by his response, but I remained calm. I acknowledged that the deadlines were tight and explained again the reasoning and the importance of having the brochure ready for the trade show. • He relaxed a little when he saw that I wasn't attacking him. He told me about all of his other competing projects and how overwhelmed he was. I asked him if there was any way that I could help him come up with a solution. • Eventually, we agreed that it would help if his manager had a better understanding of how important and time-consuming this project was. We decided we would speak with her together. • She ended up assigning some of his other projects to another designer, which took some of the pressure off of him. From <https://biginterview.com/blog/2013/09/behavioral-interview-questions-conflict.html> As a result, the designer was able to focus on the brochure and meet the deadlines. From <https://biginterview.com/blog/2013/09/behavioral-interview-questions-conflict.html>
Why leaving current company early ?	***I am currently looking for a position better matched to my skills and long-term career goals. I don't want to work just on the front end and backend but I want to work on a product where we solve critical issues regarding scalability, performance , architecture etc.***
Why amazon	1. Over the 20 years amazon has come to top by delivering products that their customer needs and solving real world problems where every engineers gets to work on critical real world software/hardware problem 2. One's work has got direct impact on millions customers. 3. Amazon has got the best minds of the world. So it would be a good career move for me Your company produces a superior product/provides a superior service. I share the values that make this possible, which should enable me to fit in and complement the team." From <https://www.thestudentroom.co.uk/showthread.php?t=855623>
what you aspire for in the life?	
What makes you satisfied? What`s the most challenging problem you have solved recently?	Being an investment banking leader and a trusted advisor since 30 years in financial firms your company provides the most trusted

Why goldman sachs ?	Over the 30 years GS has come to top by delivering products that their customer needs and solving real world problems where every engineers gets to work on critical real world software/hardware problem 1. One’s work has got direct impact on millions customers. 2. Amazon has got the best minds of the world. So it would be a good career move for me Your company produces a superior product/provides a superior service. I share the values that make this possible, which should enable me to fit in and complement the team."
Why hire you ?	A good team player

1.
2. Where do you see yourself in near future?
3. What do you want to know about GS?
4. State atleast 5 reasons why we should hire you
5. State atleast 3 reasons why Goldman Sachs is better than the company where you work

MySQL NoSql

21 September 2017 07:46

https://www.monitis.com/blog/cc-in-review-the-key-differences-between-sql-and-nosql-dbs

Topic	Links
Threading in mysql	1. https://www.xaprb.com/blog/2006/07/16/how-we-enabled-threading-in-mysql/ 2. https://www.percona.com/blog/2010/10/27/mysql-limitations-part-4-one-thread-per-connection/
Redis cluster + issues with redis cluster in terms of CAP	1. https://www.credera.com/blog/technology-insights/open-source-technology-insights/an-introduction-to-redis-cluster/
NoSQL vs MYSql scaling	1. http://www.dataversity.net/nosql-vs-sql-its-about-the-performance-and-scale/
MySQL questions	http://www.geeksforgeeks.org/commonly-asked-dbms-interview-questions/
ACID	https://en.wikipedia.org/wiki/ACID

Java questions

25 September 2017 19:51

1. OOPs concept.

1)OOPs makes development and maintenance easier where as in Procedure-oriented programming language it is not easy to manage if code grows as project size grows.

2)OOPs provides data hiding whereas in Procedure-oriented programming language a global data can be accessed from anywhere.

3)OOPs provides ability to simulate real-world event much more effectively. We can provide the solution of real word problem if we are using the Object-Oriented Programming language.

2. Abstraction vs Interface

Abstract Class	Interface	
1	An abstract class can extend only one class or one abstract class at a time	An interface can extend any number of interfaces at a time
2	An abstract class can extend another concrete (regular) class or abstract class	An interface can only extend another interface
3	An abstract class can have both abstract and concrete methods	An interface can have only abstract methods
4	In abstract class keyword “abstract” is mandatory to declare a method as an abstract	In an interface keyword “abstract” is optional to declare a method as an abstract
5	An abstract class can have protected and public abstract methods	An interface can have only have public abstract methods
6	An abstract class can have static, final or static final variable with any **access specifier**	interface can only have public static final (constant) variable

3. HashMap internal implementation.
4. Equals and Hashcode concept.
5. Set vs List
6. ArrayList vs LinkedList
7. Final keyword

JVM internal architecture.
2. Memory allocation in Java (Stack vs Heap)

Java articles

28 September 2017 22:04

String memory concepts	https://dzone.com/articles/string-memory-internals	

Remained

28 September 2017 22:40

Question: Given a remote having 0-9 digits, plus button (to increase channel), minus (to decrease) and previous channel button (to go to previous channel). We were given 2 numbers stating start and end channel number and an array having various channel numbers. The task was to go to all channel numbers given in array with minimum number of clicks.

Garbage collection in JAVA

Databases questions

30 September 2017

1. DBMS Questions- Transactions, ACID Properties, Several SQL Queries like GROUP BY, ORDER BY,
   Aggregate functions, Views Vs Tables, Triggers, Joins

Behavioral questions

06 October 2017 10:04

About goldman	The Goldman Sachs Group, Inc. is a leading global investment banking, securities and investment management firm that provides a wide range of financial services to a substantial and diversified client base that includes corporations, financial institutions, governments and individuals. Founded in 1869, the firm is headquartered in New York and maintains offices in all major financial centers around the world.
Goldman investment banking	We provide a broad range of investment banking services to a diverse group of corporations, financial institutions, investment funds and governments We aspire to be the leading trusted advisor and financier to our clients, which include corporations, financial institutions, financial sponsors, governments and public authorities and boards of directors and special committees. The Investment Banking Division (IBD) is at the front end of Goldman Sachs' client franchise. • We strive to provide best-in-class advice and execution excellence on the most complex transactions across products in order to help our clients grow. • We are focused on being a significant financier and provider of capital-raising services, which, in turn, enables our clients to achieve their strategic goals. • We remain committed to a strategy of co-investing with clients. **HOW WE ARE ORGANIZED** Our global structure allows us to better serve the strategic and financing needs of our clients across all geographies and industries. IBD encompasses two areas: IBD Classic and our Financing group. This dual structure enables us to offer the broadest range of products and advisory services, furthering our ability to deliver best-in-class solutions to our clients.
Values at goldman sachs	**OUR CLIENTS' INTERESTS ALWAYS COME FIRST.** Our experience shows that if we serve our clients well, our own success will follow. **OUR ASSETS ARE OUR PEOPLE, CAPITAL AND REPUTATION.** If any of these is ever diminished, the last is the most difficult to restore. We are dedicated to complying fully with the letter and spirit of the laws, rules and ethical principles that govern us. Our continued success depends upon unswerving adherence to this standard. **OUR GOAL IS TO PROVIDE SUPERIOR RETURNS TO OUR SHAREHOLDERS.** Profitability is critical to achieving superior returns, building our capital, and attracting and keeping our best people. Significant employee stock ownership aligns the interests of our employees and our shareholders. **WE TAKE GREAT PRIDE IN THE PROFESSIONAL QUALITY OF OUR WORK.** We have an uncompromising determination to achieve excellence in everything we undertake. Though we may be involved in a wide variety and heavy volume of activity, we would, if it came to a choice, rather be best than biggest. **WE STRESS CREATIVITY AND IMAGINATION IN EVERYTHING WE DO.** While recognizing that the old way may still be the best way, we constantly strive to find a better solution to a client's problems. We pride ourselves on having pioneered many of the practices and techniques that have become standard in the industry. **WE MAKE AN UNUSUAL EFFORT TO IDENTIFY AND RECRUIT THE VERY BEST PERSON FOR EVERY JOB.** Although our activities are measured in billions of dollars, we select our people one by one. In a service business, we know that without the best people, we cannot be the best firm. **WE OFFER OUR PEOPLE THE OPPORTUNITY TO MOVE AHEAD MORE RAPIDLY THAN IS POSSIBLE AT MOST OTHER PLACES.** Advancement depends on merit and we have yet to find the limits to the responsibility our best people are able to assume. For us to be successful, our men and women must reflect the diversity of the communities and cultures in which we operate. That means we must attract, retain and motivate people from many backgrounds and perspectives. Being diverse is not optional; it is what we must be. **WE STRESS TEAMWORK IN EVERYTHING WE DO.** While individual creativity is always encouraged, we have found that team effort often produces the best results. We have no room for those who put their personal interests ahead of the interests of the firm and its clients. **THE DEDICATION OF OUR PEOPLE TO THE FIRM AND THE INTENSE EFFORT THEY GIVE THEIR JOBS ARE GREATER THAN ONE FINDS IN MOST OTHER ORGANIZATIONS.** We think that this is an important part of our success. **WE CONSIDER OUR SIZE AN ASSET THAT WE TRY HARD TO PRESERVE.** We want to be big enough to undertake the largest project that any of our clients could contemplate, yet small enough to maintain the loyalty, the intimacy and the esprit de corps that we all treasure and that contribute greatly to our success. **WE CONSTANTLY STRIVE TO ANTICIPATE THE RAPIDLY CHANGING NEEDS OF OUR CLIENTS AND TO DEVELOP NEW SERVICES TO MEET THOSE NEEDS.** We know that the world of finance will not stand still and that complacency can lead to extinction. **WE REGULARLY RECEIVE CONFIDENTIAL INFORMATION AS PART OF OUR NORMAL CLIENT RELATIONSHIPS.** To breach a confidence or to use confidential information improperly or carelessly would be unthinkable. **OUR BUSINESS IS HIGHLY COMPETITIVE, AND WE AGGRESSIVELY SEEK TO EXPAND OUR CLIENT RELATIONSHIPS.** However, we must always be fair competitors and must never denigrate other firms. **INTEGRITY AND HONESTY ARE AT THE HEART OF OUR BUSINESS.** We expect our people to maintain high ethical standards in everything they do, both in their work for the firm and in their personal lives.

To do list

07 October 2017 11:27

1. Problems revision career cup-- done
2. Java, Spring
3. System design -done
4. Databases - MySQL Redis - done
5. Socket connections
6. Behavioral

Morgans

12 March 2018 09:49

1)types of gc , read about thread dumps , visualvm and gc commands- **remained**
2) djikstras - done
3) hashing internal working - usecases with million entries - done
4) equals and hashCode internal working - done
5) functional programming - java 8 - done
6) multithreading - read/write locks - done
7) binary tree insert search delete complexity - **remained**
8)three puzzles - eggs , pills , jars of water - **in progress**
9) elevator design - to be done
10) longest subsequence in array - **problems need revision**
11)singleton desgin pattern with synchronization , please do design patterns

1. What is singleton ? how do you make it synchronize ? can we make it thread safe without synchronize or locks ?
2. What is spring boot and spring and advantages
3. How do you handle 3 env in spring ? how does spring internally handle creating beans by env ?
4. Write code for deadlock
5. What is 12 factor app
6. What is CI
7. Whats is spring MVC how does spring converts data from rest api to ui and vice versa
8. What is java 8 memory model
9. What is volatile what are disadvantages
10. Whats is builder pattern, decorator and proxyand factory ? explain with example
11. Find duplicate characters in string
12. Print no 1to 10 without any loop.
13. Given two arrays find duplicates in o(n) time. And output it in sorted order.
14. Given name age and salary Write sql query which will give you grade column according to salary e.g salary 1000 > grade a, 10000> grade – b
15. What is transaction spring ? how does spring know that this is transaction
16. What is garbage collector how does it manage memory
17. Given one service method in java . we need to retry after every 5 minutes if service fails . write code for that.
18. What is better synchronization or locks ?
19. Complexity for treemap
20. What is microservice what are its advantages ?
21. Where should we use static and where we shouldn't ?
22. What rea types of thread pools ?
23. How does thread pool work internally ? if we pass runnables how does it ensure max no of threads ?
24. What will you do if you want to execute 20 tasks concurrently ?
25. How does junit works internally
26. How do youi keep track of how many rest api requests I got ?
27. What does @Profile do ?
28. What is DI ? how does spring works using DI ?
29. What is autowiring ?
30. How do your serialize ? when do we use it ?
31. How does threads communicate with each other ?

Morgan Stanley questions

32. How does hashmap work ?
33. I have three 3 days for operation (a+b)*(c+d) how do you make sure this happens correctly ? if I have 5 instances of each thread how do you efficiently make it thread safe?
34. Why not volatile
35. What happens if we don't override hashcode and equals
36. Write data structure and code for implementing lru cache first any complelxity then o(n) for get put delete and search
37. How are strings stored in memory ? how object references work ?
38. How would you sort on name and age on employee class ? whats the difference between comparator and comparable . why cant we use comparable for sorting. Treempa complexity
39. Write code for above
40. When would you use hashmap and when treemap
41. What re ways of starting threads ?
42. Write code for reversing string with recursion and one input parameter
43. We have 100 files with name and skills write program to get the name against list of skills in minimal time. How you make sure operations are thread safe ?
44. Write classes for furniture like plastic wooden metal furniture having chair, bed, table, and dimensions like top side bottom and calculate weight based on type and furniture
45. How do you know when thread is terminated either normally or abruptly
46. If I don't override hashcode and equals and put object in map will I be able to get it ?
47. How are buckets stored ? whats the worst case for hashmap ?
48. What is difference between hashmap concurrent hashmap and synchhronizedMap?
49. What is stringbuffer ? where does it fails ?
50. I want to write exact copy of object but with different address how will I do it ? do we need to override clone method after implementing clonable ? if not where does that method come from ?
51. What is set ? how does it ensure non duplicates ?
52. What immutable object ? write code . where is it used ? what are advantages
53. Given a table with emp_id, name, manager_id where manager_id references to same table emp_id Write sql query to get employee names and its manager's name .
54. What re types pf exception ? how do you handle those ? errors , runtime and checked ? when do we use throws ? is it mandatory to throw or catch runtime?
55. Given a matrix find no of ways to reach end of matrix for (0,0).
56. Write code for priority queue with minimal complexity for getMax getMin search delete and add
57. You have stream of integers coming find median. E.g 1 4 arrive median – 4 , 5 arrived median = 4.5 3 arrived ==median – 3, 1 arrived median- 3.5
58. Design architecture for showing stock prices of companies in a grid and where user can buy stocks in few seconds
59. What is ownership?
60. If some other code has bug what wil, you do ?
61. What do you unlike and like in your manager
62. I have two service instance which talks to a database and both write and read at same time how to make data consistent? If one server crashes what will you do ?
63. Long discussion on microservice and modularity
64. What if your managers tell you to work on something you don't like ? what would you do ?
65. What are your strengths ?
66. What do you do to succeed in any organization ?
    What qualities you need to succeed ?