**Flipkart telephonic - 0.5 hour**

* Explain one of your projects from resume
* Question: Given a very large file containing (x,y) coordinates - the file cannot fit into memory, select the K coordinates which are closest to the origin (0,0). Describe approach and time complexity?
  + Ans: Select first K and keep them in a priority queue, every time you get a new one, if its distance is less than the one having max distance in the queue, delete max, insert new. Repeat until last. nlogK solution.

**Flipkart onsite Round 1: Machine Coding**

Q. Code a pubsub service with a broker which gives the following guarantees:

* Guaranteed delivery of messages to each subscriber at least once
* Each subscriber should be able to specify their own pace of consuming messages concurrently. Eg: publisher can publish 100 per second but S1 can consume only 20 per second concurrently, provide for that
* Durability of messages

Ans:

* For durability and guaranteed delivery, we need persistence of all messages
* For guaranteed delivery, we need acking mechanism
* For concurrency concerns we need to be able to allow each subscriber to have his own thread pool which consumes messages
* After discussing solution with persistence layer, code the in memory solution ensuring it is extendable to persistence
* In a broker, I kept one queue of messages per topic as they are coming from producers
* Have a worker which posts messages from main queue to per subscriber queue of messages, message can be deleted from main queue once they are in subscriber’s queue
* A thread pool of workers (number specified by subscriber) for each subscriber can consume messages as and when they arrive in its queue and post them to whatever endpoint it needs to be posted to (for simplicity, it is being posted to system output)
* Make sure the broker exposes a publish interface for the producers to use
* Broker and Subscriber have test methods called by main method to demo how this will work

I was asked to email the code after 1.5 hours to the interviewer. No one came to discuss it with me.

**Flipkart onsite Round 2: Problem Solving - 1 Hour**

Q. Given a list of integer ranges, find the number of distinct integers present in the union of all the ranges. Eg:

For ranges: 1-2, 3-4, 0-10, output is 11 (all integers from 0 to 10 are present)

Ans: Merge all the ranges and get a list of non overlapping ranges, for each range add end-start+1 in the final answer

Q. What is the space and time complexity? And how can we optimise this

Ans: Time complexity is O(nlogn) as we need to sort the intervals to be able to merge them. Space is O(n) worst case as we may end up creating another list of n intervals if none overlap

Optimise space by not storing any non overlapping intervals and instead getting the end-start+1 and adding it while merging

Q. Given n buckets, each having a minimum capacity and a maximum capacity, buckets cannot store stones less than min and more than max capacity. You can choose any subset of this buckets and keep stones in them such that they fulfill each buckets condition. The sum of these stones is K. How many such K’s are possible?

Ans: This problem can be reduced into the above problem of merging intervals.

We have to consider all subsets of n buckets - 2^n subsets are possible

If subset is of size 1: K can be anything from min to max for that bucket

If subset of size 2: K can be anything from sum(all mins) to sum(all max) for the buckets in the subset

This is provable for all subsets

So, this problem is now, finding all subsets of the given set of buckets, and finding the ranges from sum(min) to sum(max) which will give us 2^n intervals and then we need to merge overlapping intervals and find the count of numbers in the union

**Flipkart onsite Round 3: System Design - 1 Hour**

Some discussion about my most interesting project and its issues

Q. On any ecommerce website, when a product is out of stock, there is a notify me option. Design the system backing this feature, which takes a notify me intent, and when a product comes back in stock, it send notifications to all the users registered

Follow up questions:

* How will you handle million users needing to be sent same notification when only very few pieces of the product are available
  + Think about ranking of users
* How will you design the DB schema, detailed discussion into this
* RDBMS vs NoSQL
* How will you shard the RDBMS schema, pros and cons
* How will you make sure same notifications is not sent to the user more than once