INTERVIEW
KICKSTART

TECHNICAL INTERVIEW CHECKLIST

Technical Interview Checklist: The Most Comprehensive Prep Checklist to Nail Coding Interviews in 2021

Interview Kickstart offers the best technical interview prep courses that make you a better engineer and help you nail tech interviews. Since 2014, we have **trained over 6200 experienced** software engineers. Our alums have landed **dream jobs** at Facebook, Apple, Amazon, Netflix, Google, and many more top tech companies.

The highest compensation received by an IK alum is a whopping **$933,000!**

From experience, we know that cracking tech interviews is not about practicing an insane number of problems. Instead, these problems can be organized and distilled into much fewer topics. In this checklist, we’re sharing a bulk of that organization with you.

To learn more about us, visit www.interviewkickstart.com

Must-Learn Topics for Coding Interviews

□ Basic math

Relevant parts of discrete math pertaining to combinatorics

- □ Algebra (linear and quadratic equations, arithmetic, and geometric series)
- □ Combinatorics
- □ Recursive mathematical functions
- □ Proofs by mathematical induction
- □ Decrease and conquer
- □ Asymptotic analysis

□ Basic data structures

For storing a collection of “n” like items

- □ Arrays
- □ Linked lists
- □ Stacks
- □ Queues and deques
- □ Linear search
- □ Binary search
- □ Binary search trees
- □ Hash tables

□ Bit manipulation

- □ Conversion from base 10 to base 2 and vice versa
- □ Finite (32 bit) representation of an infinite number line
- □ Representing negative numbers (using 2s complement, Boolean operators)
- □ Multiplication and division
- □ Other data types (floating point, character encodings)

□ Binary search variants

- □ Regular binary search
- □ Bisection
- □ Binary search for optimization

□ Sorting algorithms

- □ Quicksort
- □ Merge sort
- □ Heap sort
- □ Bubble sort
- □ Selection sort
- □ Insertion sort
- □ Counting sort
- □ Radix sort
- □ Bucket sort
- □ Cycle sort

□ Extensions of merge sort

Two-pointer pass in two arrays

□ Extensions quicksort

- □ Quickselect pattern
- □ Three-way partitioning pattern

□ Two-sum pattern

Presorting vs. hash tables

□ Selection in a stream using heaps

□ Interval line sweep

□ Linked lists

- □ Floyd Cycle detection
- □ Sorting and partitioning
- □ List reversal

- □ **Generic decrease and conquer for array problems**

- □ **Prefix sum**

- □ **Sliding windows**
  - □ Fixed-length windows
  - □ Variable-length windows

- □ **Combinatorial enumeration**

- □ **Backtracking**

- □ **Tree traversal patterns**
  - □ BFS
  - □ DFS
  - □ Top-down
  - □ Bottom-up
  - □ Boundary walk
  - □ Iterative

- □ **Tree construction patterns**

- □ **Graphs foundation**

  Graph theory
  - □ BFS/DFS on undirected graphs
  - □ BFS/DFS on directed graphs
  - □ BFS/DFS on 2D grids

- □ **Dynamic programming (DP)**
  - □ DP on sequences
  - □ DP on sub-trees
  - □ DP on permutations
  - □ DP on subsets
  - □ DP on two-strings

□ Greedy algorithms foundations with interval problems

□ Advanced graphs

- □ Bridges and articulation points
- □ Strongly connected components (Tarjan, Kosaraju)
- □ Union-find foundations and coding pattern
- □ Eulerian path construction
- □ Combinatorial optimization on graphs
- □ Shortest-path problem
- □ Minimum spanning trees
- □ All-pairs shortest paths
- □ State-space tree
- □ Graph search

□ Advanced trees

- □ AVL
- □ Red-black
- □ Segment
- □ Binary-indexed
- □ B-trees
- □ Quad trees

□ Pattern matching

- □ KMP
- □ Rabin Karp
- □ Tries

□ Ad-hoc problems

Such as design skip lists

Must-Learn Topics for Systems Design Interviews

□ Basics of systems design

- □ Online Processing
- □ Batch Processing
- □ Stream Processing

□ Basics of networking

- □ Network protocols
- □ Webserver
- □ Cryptographic hash functions

□ Scaling distributed applications

- □ Reasons of scaling (data size, throughput, fault tolerance, geolocation and hotspots)
- □ Horizontal scaling
- □ Vertical scaling
- □ Load balancing
- □ Server proxy (reverse and forward)
- □ CAP theorem
- □ Content distribution networks

□ Replication

- □ Single leader
- □ Multileader
- □ Leaderless

□ Sharding techniques

- □ Partitioning vs. replication
- □ Partitioning of key value data
- □ Partitioning and secondary indexes
- □ Rebalancing partitions

□ Measuring the performance of scalable system

- □ Performance metrics of a scalable system
  - □ Correctness
  - □ Availability
  - □ Throughput
  - □ Response time
- □ Service-level agreements

□ Cache

- □ Reads and writes
- □ LRU cache
- □ Strategies
- □ Consistent hashing

□ Storage and retrieval

- □ Key-value stores
- □ Relational database and tree index
- □ SQL, normalization, and keys
- □ ACID transactions
- □ Big data
- □ NoSQL

□ MapReduce and distributed file systems

- □ MapReduce Framework
- □ Distributed file system

□ Searching in a corpus of documents

- □ Inverted index
- □ External sort merge
- □ K-way external sort-merge
- □ Distributed sorting

□ Systems design case studies

- □ URL shortener
- □ Streaming services
- □ Chat messenger server
- □ Recommendation system
- □ Maps
- □ Search Engine
- □ Unique ID generator

□ Object modeling

Not required for all companies

- □ Basics of UML
- □ Design patterns
  - □ Composite pattern
  - □ Decorator pattern
  - □ Facade pattern
  - □ Visitor pattern
  - □ Flyweight pattern
  - □ Proxy pattern
  - □ Command pattern
  - □ Observer pattern
  - □ Strategy pattern
  - □ State pattern
  - □ Factory pattern
  - □ Singleton pattern

□ Basics of API design

- □ RESTful API design
- □ SOLID principles

□ Concurrency

Not required for all companies

- □ Parallelism vs. concurrency
- □ Blocked vs. running
- □ Mutex
- □ Cross-process mutex
- □ Condition variable
- □ Semaphore
- □ Atomic operations
- □ Deadlock

Note: This list is more aligned to core software engineering roles. If you come from a more specialized domain, such as data engineering, you only need a subset of these topics. However, in such specialized domains, you also need specialized courses to interview-hone your skills, which Interview Kickstart offers separately. Join our free webinar to learn more.