Amazon 6m internship question bank

By Harine Vidyasekaran - AVV Chennai, AWS Cloud captain

This question bank contains questions from both on campus as well as off campus Amazon interviews. 98% of the questions are present on LeetCode and the other 2% can be found on GFG. None of the questions are newly innovated or proprietary to Amazon, they are classic DSA questions. By following Neetcode 150 & Strivers SDE sheet one can cover most of the patterns in these questions. Here, the majority of the questions are medium level. Amazon also likes asking sliding window and binary tree questions more often.

BASIC DETAILS:

1. Most interviews were 60-70 minutes long if conducted off campus and exactly 1 hour long or even lesser if conducted on campus
2. (Mostly) Must solve at least 2 coding questions per interview to pass
3. Interviewers expected approach explanation before coding
4. A brute force / naive approach is expected to be stated first, with a higher time complexity, and then the optimal approach must be stated, with a lower time complexity.
5. Must know time, space complexity of approach, it is often noted down by the interviewer
6. Live code (for typing code) & Amazon Chime (for the video call) are the platforms used
7. 2 or more employees will attend the interview, where the main employee conversing with you will stay with you on the Live code platform and paste the question, explain it and guide you with hints if needed. The other employees check eyeball movement and if you're attending the interview ethically or not.

Data structures covered: Arrays (sliding window, 2 pointer, stack, queue), Strings, Linked lists, Binary trees & BST (most occurring), heaps, graphs and DP.

ARRAYS – GENERAL:

- Maximize sum of array after K negations (https://leetcode.com/problems/maximize-sum-of-array-after-k-negations/)
- Rotate matrix by 90deg (https://leetcode.com/problems/rotate-image/description/)
- Next greater element (https://leetcode.com/problems/next-greater-element-i/description/)
- Jump game (https://leetcode.com/problems/jump-game/description/)
- Merging intervals (https://www.geeksforgeeks.org/merging-intervals/)
- Make array strictly increasing (https://www.geeksforgeeks.org/check-whether-an-array-can-be-made-strictly-increasing-by-removing-at-most-one-element/)
- Asteroid Collision (https://leetcode.com/problems/asteroid-collision/)

ARRAYS - SLIDING WINDOW / TWO POINTER:

- Maximum subarray sum (Kadane's Algorithm) (https://leetcode.com/problems/maximum-subarray/)
- Fruit in a basket (https://leetcode.com/problems/fruit-into-baskets/description/)
- 3Sum (https://leetcode.com/problems/3sum/)
- Trapping rainwater (https://leetcode.com/problems/trapping-rain-water/submissions/1460631178/)
- Search in rotated sorted array (https://leetcode.com/problems/search-in-rotated-sorted-array/description/)
- 2Sum II (https://leetcode.com/problems/two-sum-ii-input-array-is-sorted/description/)

STRINGS:

- Remove adjacent duplicates (https://www.geeksforgeeks.org/python-program-to-recursively-remove-all-adjacent-duplicates/)
- Reorganize string (https://leetcode.com/problems/reorganize-string/description/)

LINKED LISTS:

- Linked list cycle (https://leetcode.com/problems/linked-list-cycle/)
- Find intersection/junction of two linked lists (https://leetcode.com/problems/intersection-of-two-linked-lists/)

BINARY TREES & BST:

- Binary tree zigzag traversal (https://leetcode.com/problems/binary-tree-zigzag-level-order-traversal/)
- Minimum depth of binary tree (https://leetcode.com/problems/minimum-depth-of-binary-tree/description/)
- Bottom view of binary tree (https://takeuforward.org/data-structure/bottom-view-of-a-binary-tree/)
- Maximum path sum (https://leetcode.com/problems/binary-tree-maximum-path-sum/description/)
- Lowest Common Ancestor (LCA) (https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-tree/description/)
- Create BST from string (https://www.geeksforgeeks.org/construct-binary-tree-string-bracket-representation/)
- Two Sum in BST (https://leetcode.com/problems/two-sum-iv-input-is-a-bst/)
- Minimum distance between 2 nodes in a BST (https://leetcode.com/problems/minimum-distance-between-bst-nodes/description/)
- Maximum element between 2 nodes BST (https://www.geeksforgeeks.org/maximum-element-two-nodes-bst/)

HEAPS:

- Kth largest element in array (https://leetcode.com/problems/kth-largest-element-in-an-array/description/)
- Connect n ropes minimum cost (https://www.geeksforgeeks.org/connect-n-ropes-minimum-cost/)
- Farthest building you can reach (https://leetcode.com/problems/furthest-building-you-can-reach/description/)
- Median of Data Stream (https://leetcode.com/problems/find-median-from-data-stream/)

GRAPHS:

- Find all paths between 2 nodes (Graph traversal)
- Number of islands (https://leetcode.com/problems/number-of-islands/description/)
- Shortest path to reach end of grid (https://leetcode.com/problems/shortest-path-in-a-grid-with-obstacles-elimination/description/)
- Rat in a maze (https://takeuforward.org/data-structure/rat-in-a-maze/)
- Course scheduling (https://leetcode.com/problems/course-schedule/description/)
- Rotting Oranges (https://leetcode.com/problems/rotting-oranges/)

DYNAMIC PROGRAMMING:

- Max profit in job scheduling (https://leetcode.com/problems/maximum-profit-in-job-scheduling/description/)
- Coin change (https://leetcode.com/problems/coin-change/description/)
- Max path sum with obstacles (2D DP - Hard)