



Советы для успешного прохождения собеседований:

- Потренируйтесь решать задачи на онлайн-ресурсах;
- Проговаривайте вслух ход мыслей, подробно обосновывайте выводы. - Эксперту важно не только получить правильное решение задачи, но и понять, каким путем вы к нему пришли;
- Не стесняйтесь просить подсказку у интервьюера;
- Обязательно проверяйте код на ошибки, перед тем как его сдать;
- Старайтесь предлагать разные решения: первое решение не всегда самое удачное. Если вам пришло в голову, как его можно улучшить, — непременно скажите об этом, даже если задачу уже рассмотрели и отложили в сторону.



Алгоритмы, которые чаще всего бывают в задачах:

- сортировки (например, bubble sort или quicksort);
- разворота одно/двусвязного списка;
- разворота строки
- обхода дерева



Материалы для подготовки

1. Система Яндекс.Контест, где стоит потренироваться решать задачи, т.к. они схожи с теми, которые дают на собеседовании:
<https://contest.yandex.ru/contest/8458/enter/>
2. Система Яндекс.Код, которую используют при решении задач на ноутбуке:
<https://code.yandex-team.ru/>
3. Можно попрактиковаться в алгоритмах и порешать задачи на leetcode / hackerrank / geeksforgeeks, например:
<https://leetcode.com/problemset/algorithms/>
4. Оценка алгоритмов:
<https://habr.com/ru/post/188010/>
5. В этих видео мы разбираем решение алгоритмических задач:
<https://www.youtube.com/watch?v=0yxjWwoZtLw>
<https://www.youtube.com/watch?v=zU-LndSG5RE>

Темы и ссылки, где можно лучше подготовиться к алгоритмам:

linked lists:

- <https://leetcode.com/problems/merge-k-sorted-lists/>
- <https://leetcode.com/problems/linked-list-cycle/>
- <https://leetcode.com/problems/add-two-numbers/>
- <https://leetcode.com/problems/reverse-linked-list/>

binary search:

<https://leetcode.com/problems/binary-search/>
<https://leetcode.com/problems/guess-number-higher-or-lower/>
<https://leetcode.com/problems/search-a-2d-matrix/>
<https://leetcode.com/problems/search-in-rotated-sorted-array/>
<https://leetcode.com/problems/find-minimum-in-rotated-sorted-array/>
<https://leetcode.com/problems/search-in-rotated-sorted-array-ii/>

hash table:

<https://leetcode.com/problems/single-number/> - решить за $O(1)$ по памяти
<https://leetcode.com/problems/two-sum/>
<https://leetcode.com/problems/4sum/>
<https://leetcode.com/problems/group-anagrams/>
<https://leetcode.com/problems/valid-anagram/>
<https://leetcode.com/problems/find-all-anagrams-in-a-string/>

queue/stack:

<https://leetcode.com/problems/valid-parentheses/>

dfs/bfs:

<https://leetcode.com/problems/number-of-islands/>
<https://leetcode.com/problems/remove-invalid-parentheses/>

sort:

<https://leetcode.com/problems/merge-intervals/>

heap/hash:

<https://leetcode.com/problems/top-k-frequent-words/>
<https://leetcode.com/problems/top-k-frequent-elements/>

two pointers:

<https://leetcode.com/problems/container-with-most-water/>
<https://leetcode.com/problems/partition-labels/>

sliding window:

<https://leetcode.com/problems/sliding-window-median/>
<https://leetcode.com/problems/sliding-window-maximum/>
<https://leetcode.com/problems/longest-repeating-character-replacement/>

tree:

<https://leetcode.com/problems/same-tree/>
<https://leetcode.com/problems/symmetric-tree/>
<https://leetcode.com/problems/balanced-binary-tree/>
<https://leetcode.com/problems/path-sum-ii/>

greedy problems:

<https://leetcode.com/problems/best-time-to-buy-and-sell-stock/>
<https://leetcode.com/problems/best-time-to-buy-and-sell-stock-ii/>
<https://leetcode.com/problems/best-time-to-buy-and-sell-stock-with-transaction-fee/>
<https://leetcode.com/problems/best-time-to-buy-and-sell-stock-with-cooldown/>