**Data Structures and Algorithms Resources**

[DSA resources | IIITV Open Source](https://iiitv.github.io/resources/dsa-cp.html)

**Step 1: Learn a Programming language (C, C++ or Java)**

[**C Language Tutorial**](https://www.youtube.com/watch?v=EMEvheCVhMk&list=PL7ersPsTyYt2Q-SqZxTA1D-melSfqBRMW)

* <https://www.youtube.com/playlist?list=PL7ersPsTyYt2Q-SqZxTA1D-melSfqBRMW>

**C++ Language Tutorial**

* <https://www.youtube.com/playlist?list=PLu0W_9lII9agpFUAlPFe_VNSlXW5uE0YL>

**Java Language Tutorial**

* <https://www.youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q>

**Step 2: Understand Basic Concepts**

**Asymptotic notations**

* **Video tutorials**
* <https://www.iarcs.org.in/inoi/online-study-material/topics/efficiency.php>
* <https://www.youtube.com/watch?v=9TlHvipP5yA&list=PLDN4rrl48XKpZkf03iYFl-O29szjTrs_O&index=6>
* **Article**
* <http://www.iitk.ac.in/esc101/08Jul/lecnotes/practise_sol.pdf>
* <https://runestone.academy/runestone/books/published/pythonds/AlgorithmAnalysis/toctree.html>

**Number Theory Lectures**

* **Video tutorials**
* <https://www.youtube.com/playlist?list=PLauivoElc3giVROwL-6g9hO-LlSen_NaV>
* <https://www.youtube.com/playlist?list=PL2q4fbVm1Ik4liHX78IRslXzUr8z5QxsG>
* **Article**
* <https://cp-algorithms.com/>

**Sorting Algorithms**

* **Video tutorials**
* <https://visualgo.net/en/sorting?slide=1>
* <https://www.youtube.com/playlist?list=PLsFNQxKNzefK_DAUwnQwBizOmcY7aDLoY>
* **Article**
* <https://www.geeksforgeeks.org/sorting-algorithms/>
* <https://www.iarcs.org.in/inoi/online-study-material/topics/sorting.php>

**Searching Algorithms**

* **Video tutorials**
* <https://visualgo.net/en/dfsbfs?slide=1>
* <https://www.youtube.com/playlist?list=PLauivoElc3gjE_s-7owHO0RVb_jj7Rx85>
* **Article**
* <https://www.geeksforgeeks.org/searching-algorithms/>

**Recursion**

* **Video Tutorial**
* <https://www.youtube.com/playlist?list=PL_z_8CaSLPWeT1ffjiImo0sYTcnLzo-wY>
* **Article**
* <https://www.csee.umbc.edu/~chang/cs202.f98/readings/recursion.html>
* <https://www.byte-by-byte.com/recursion/>

**LinkedList**

* **Video Tutorial**
* <https://www.youtube.com/playlist?list=PL-Jc9J83PIiF5VZmktfqW6WVU1pxBF6l_>
* **Article**
* <https://www.hackerearth.com/practice/data-structures/linked-list/singly-linked-list/tutorial/>

**Stack & Queue**

* **Video Tutorial**
* <https://www.youtube.com/playlist?list=PL-Jc9J83PIiEyUGT3S8zPdTMYojwZPLUM>
* **Article**
* <https://www.hackerearth.com/practice/notes/stacks-and-queues/>

**Greedy Algorithms**

* **Video tutorials**
* **Article**
* <https://www.iarcs.org.in/inoi/online-study-material/topics/greedy.php>
* <https://www.topcoder.com/thrive/articles/Greedy%20is%20Good>

**Bitmanipulation**

* **Video tutorials**
* <https://www.youtube.com/playlist?list=PL2q4fbVm1Ik7ip1VkWwe5U_CEb93vw6Iu>
* **Article**
* <https://www.hackerearth.com/practice/notes/bit-manipulation/>

**Dynamic Programming :**

* **Video tutorials**
* [Dynamic Programming by Aditya Verma Playlist](https://www.youtube.com/playlist?list=PL_z_8CaSLPWekqhdCPmFohncHwz8TY2Go)
* **Article**
* <https://www.freecodecamp.org/news/demystifying-dynamic-programming-3efafb8d4296/>
* <https://www.topcoder.com/thrive/articles/Dynamic%20Programming:%20From%20Novice%20to%20Advanced>

**Trees :**

* **Video tutorials**
* <https://www.youtube.com/playlist?list=PLgUwDviBIf0q8Hkd7bK2Bpryj2xVJk8Vk>
* **Article**
* <https://www.iarcs.org.in/inoi/online-study-material/topics/dp-trees.php>
* <https://people.eecs.berkeley.edu/~vazirani/s99cs170/notes/dynamic2.pdf>

**Graphs :**

* **Video tutorials**
* <https://www.youtube.com/playlist?list=PLgUwDviBIf0rGEWe64KWas0Nryn7SCRWw>
* **Article**
* <https://www.iarcs.org.in/inoi/online-study-material/topics/graphs.php>

**Segment Trees:**

* **Video tutorials**
* <https://www.youtube.com/playlist?list=PL2q4fbVm1Ik6v2-emg_JGcC9v2v2YTbvq>
* **Article**
* <https://cp-algorithms.com/data_structures/segment_tree.html>

**Step 3: Practice questions**

**Platforms:**

* [GeeksforGeek](https://practice.geeksforgeeks.org/topic-tags/)s
* [Leetcode](https://leetcode.com/problemset/all/)
* <https://www.codechef.com/>
* <https://codeforces.com/>

**Practice Sheet:**

* [DSA 450 Questions Sheet - Love Babbar](https://450dsa.com/)
* [A2OJ Ladders](https://a2oj.com/Ladders.html)
* [cp-contests | 💻 Archive of Competitive Programming Contests hosted by IIITV CC](https://iiitv.github.io/cp-contests/)