

CodeZoned Computer Science Resources

Introduction to Computer Science

Introduction Course: <https://www.edx.org/course/cs50s-introduction-computer-science-harvardx-cs50x>

Important Links:

1. **Stanford Library:** <http://cslibrary.stanford.edu/>
2. **Programming Camp Syllabus:**
https://docs.google.com/document/d/1_dc3Ifg7Gg1LxhiqMMmE9UbTsXpdRiYh4pKILYG2eA4/edit
3. **Books:** https://drive.google.com/folderview?id=1TYisa-3_A8AeAdfBFAGrEMrb69ynibpu ,
<https://www.google.co.in/amp/s/www.webpagefx.com/blog/web-design/free-books-code/amp/> ,
<https://www.onlineprogrammingbooks.com>
4. **Academic Torrents:**
<http://academictorrents.com>
<http://academictorrents.com/collection/video-lectures>
5. **Online Programming Courses and Tutorials:**
<http://hackr.io>
6. **OSSU for CSE:**
<https://github.com/ossu/computer-science/blob/dev/README.md>
7. **Programming books for free:**
<https://goalkicker.com>

Python

Introductory courses and tutorials:

MIT OCW:

<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/lecture-videos/>

EdX:

<https://www.edx.org/course/introduction-python-absolute-beginner-microsoft-dev236x-0>

Tutorials:

<https://docs.python.org/3/tutorial/>

Full stack python:

<https://drive.google.com/file/d/0B6KlugcejrSMVTZCNVEzUWJKMjg/view?usp=drivesdk>

Data Structures & Algorithms

Introductory courses and tutorials:**MIT OCW**

<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos/>

IISc:

<http://lcm.csa.iisc.ernet.in/dsa/dsa.html>

Miscellaneous:

<http://e-maxx.ru/>

<https://discuss.codechef.com/questions/64426/awesome-resource-for-ds-and-algorithms>

<https://www.technotification.com/2018/03/top-10-algorithms-for-programmer.html/amp>

<https://www.youtube.com/user/mycodeschool/playlists>

Sieve of Eratosthenes

<https://www.geeksforgeeks.org/sieve-of-eratosthenes/>

AVL & Red Black Trees:

<https://www.geeksforgeeks.org/avl-tree-set-1-insertion/>

http://btechsmartclass.com/DS/U5_T3.html

http://btechsmartclass.com/DS/U5_T4.html

<https://www.geeksforgeeks.org/red-black-tree-set-1-introduction-2/>

Merge Sort Tree:

<https://discuss.codechef.com/questions/94448/merge-sort-tree-tutorial>

Heaps:

<https://www.hackerearth.com/practice/data-structures/trees/heapspriority-queues/tutorial/>

MO's Algorithm:

<http://codeforces.com/blog/entry/7383>

<https://blog.anudeep2011.com/mos-algorithm/>

Knapsack Problem:

https://en.m.wikipedia.org/wiki/Knapsack_problem

Big O cheat sheet:

<http://bigochaatsheet.com/pdf/big-o-cheatsheet.pdf>

Practice Algorithms:

<http://hackerearth.com/practice/algorithms>

Problem Solving with DSA and Python:

<http://interactivepython.org/courselib/static/pythonds/index.html>

Visualizing Algorithms:

<https://visualgo.net/en>

<http://sorting.at/>

Competitive Programming

Introduction to CP:

<https://www.hackerearth.com/practice/notes/getting-started-with-the-sport-of-programming/>

<https://discuss.codechef.com/questions/37684/learn-competitive-programming>

<https://discuss.codechef.com/questions/18752/what-are-the-must-known-algorithms-for-online-programming-contests>

Related Materials:

<https://github.com/Inishan/awesome-competitive-programming/blob/master/README.md>

Square Root Decomposition:

<https://m.youtube.com/watch?v=VGq6w9TIJBY>

Floating Point Arithmetic:

https://docs.oracle.com/cd/E19957-01/806-3568/ncg_goldberg.html

Tools to improve programming skills:

<https://www.technotification.com/2018/04/tools-improve-programming-skills.html/amp>

Standard Template Library:

<https://www.topcoder.com/community/data-science/data-science-tutorials/power-up-c-with-the-standard-template-library-part-1>

Python v/s C:

<https://www.quora.com/Should-I-begin-with-C%2B%2B-or-python-in-competitive-programming/answer/Mariya->

Mykhailova?share=454ed0dd&srid=hTaTH
<https://softwareengineering.stackexchange.com/questions/125576/why-is-c-predominant-in-programming-contests-and-competitions>

Discrete Mathematics

Introductory Courses:

MIT OCW:

<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/>

Generating Functions:

http://cse.iitkgp.ac.in/~animeshm/generating_funct.pdf

Sequences and Series:

https://en.wikipedia.org/wiki/Category:Sequences_and_series
<https://oeis.org/>

Famous Theorems & Conjectures:

Four Color Theorem:

https://en.m.wikipedia.org/wiki/Four_color_theorem

Goldbach's Conjecture:

https://en.m.wikipedia.org/wiki/Goldbach%27s_conjecture

Godel's Incompleteness Theorems:

https://en.m.wikipedia.org/wiki/Gödel%27s_incompleteness_theorems

Menger's Theorem:

https://en.m.wikipedia.org/wiki/Menger%27s_theorem

Developmental Coding

XHTML and CSS Tutorials:

https://youtu.be/cqszz_OfAFQ

Android Tutorials Playlist:

http://www.youtube.com/playlist?list=PLlxmoA0rQ-LyCGSSD_nuPAmXDSR_FU0RR

Algorithm and Data Structures Course:

Necessary Stuff:

https://drive.google.com/drive/folders/0ByW00aO1eI_MN1BEd3VNRUZENkU

Web development with a head start:

<https://www.geeksforgeeks.org/begin-web-development-with-a-head-start/>

<https://www.freelock.com/newsletter/10-problems-web-development-projects-and-how-weve-solved-them>

Coding Style and its importance:

<https://www.smashingmagazine.com/2012/10/why-coding-style-matters/>

<https://users.ece.cmu.edu/~eno/coding/CCodingStandard.html>

<https://javascript.info/coding-style>

Important Git Commands:

<https://www.technotification.com/2018/04/important-git-commands.html/amp>

Git and Github Live Webinar:

<https://youtu.be/D3RVdbICmk0>

Machine Learning

Pre-requisites:

<http://www.sharpsightlabs.com/blog/machine-learning-prerequisite-isnt-math/>

<https://www.quora.com/What-are-prerequisites-to-start-learning-machine-learning>

<https://elitedatascience.com/learn-machine-learning>

Linear Algebra:

<https://bit.ly/2JjHsXA>

Calculus:

<http://bit.ly/gilbert-strang-calculus>

Optimization Methods:

<http://bit.ly/EDX-MIT-Optimization-Methods>

Probability and Statistics:

Probability distributions:

<http://mathlets.org/mathlets/probability-distributions/>

<http://bit.ly/MIT-Applied-Probability>

<http://bit.ly/Statistics-Part-1>

<http://bit.ly/Statistics-Part-2>

Learning Path:

<https://gist.github.com/hardik2396/83d642af3b22811fac6719bf28ceb048>

<https://gist.github.com/kdexd/4b41e6edbbffb9886b1b2a121d327b1d>

Weka v/s Scikit:

<https://www.quora.com/What-is-a-better-machine-learning-library-Weka-or-scikit-learn?share=e99bd97c&srid=uxA0R>

Top-5 challenges in ML:

<https://youtu.be/AzDy55uhY3o>

Computer Networks

Computer Networks Courses:

<https://www.cse.iitk.ac.in/users/dheeraj/cs425/>

<https://www.codeproject.com/Articles/1232042/Introduction-to-Convolutional-Neural-Networks>

How DNS works?

<http://howdns.works>

Ethical Hacking:

<https://www.cybrary.it/course/ethical-hacking-scratch/>

Symbolic Computation

Chebyshev Polynomials:

https://en.m.wikipedia.org/wiki/Chebyshev_polynomials

Algorithms for Computer Algebra:

<http://download1.libgen.io/ads.php?md5=343FE31EFD7105329DB99CD54C702528>

Miscellaneous

MIT OCW course list:

<https://ocw.mit.edu/courses/>

Coder's Diary:

<https://play.google.com/store/apps/details?id=asquero.com.myapplication>

Dart introduction:

<https://youtu.be/5KInlCq2M5Q>

How to pass a function as a parameter in C:

<https://stackoverflow.com/questions/9410/how-do-you-pass-a-function-as-a-parameter-in-c>

