

Computer Science Society

Resources for Computer Science Students at **Cal Poly Pomona**.

CPP-Related:

[Cal Poly Pomona: Computer Science](#) - The official page for the Computer Science department.

[Computer Science Society](#) - The official wordpress for the Computer Science Society club (may be down)

[CSS Official Facebook Group](#) - A great way to keep in touch and network with other CS majors since everyone has a facebook.

[Friends of CS @ Cal Poly Pomona](#) - A linkedin group created by Prof. Jaspan to help students, professors, and alumni keep in touch

[CSS Official Freenode IRC Channel](#) - Just a link to freenode's web IRC client. The channel is '#calpolycss'. Supply a nick and you're ready to go.

General:

[Coding Horror](#) - A humorous blog for programmers and developers.

[TopCoder](#) - A great resource for programmers; has software dev contests as well as tutorials and articles.

[What All CS Majors Should Know](#) - An article written by a Ph.D. student explaining what all CS majors should know by the end of their undergraduate studies.

Subreddits: [/r/compsci](#), [/r/math](#), [/r/algorithms](#), [/r/learnprogramming](#) - Yeah, I know it's reddit, but it's a great resource if you're bored.

[Quora](#) - A great place to ask questions related to the field that are not too technical.

Videos:

[MIT Lectures \(Youtube\)](#) - A great additional resource to use if you don't understand what's going on in your class.

[The New Boston](#) - Tutorials for basic programming and science courses.

[Khan Academy](#) - A Python and Javascript approach to CS.

Note: there are two different CS categories, one is under Science & Economics (Python) and the other has its own Computer Science listing (Javascript).

Programming:

[Stackoverflow](#) - A great community to ask for programming-related questions.

[Codecademy](#) - Learn how to code in a simple and interactive way. No IDE is needed. Currently focuses on scripting.

[Lucidchart](#) - For diagramming and rapid prototyping of wireframes and mockups. Very useful for planning large projects.

Mobile:

[200-Episode Android Tutorials](#) - A spectacular series about getting started in Android development, over 20 hours of videos. **Caution: some information might be outdated.**

Theory:

[wikipedia](#) - an excellent resource for well-moderated concepts such as [graph theory](#), [recursion](#), [mathematical induction](#), etc. A page is typically well moderated if it has many subsections and substantial sources.

Practice/Challenge Your Craft:

[TopCoder](#) - Problem solving using programming.

[ProjectEuler](#) - Number theory problem solving using programming.

[Rosalind](#) - Bioinformatics problem solving.

[Google Code Jam](#) - An annual competition hosted by Google. You can practice problems from previous years.

[Reddit's Daily Programmer](#) - A subreddit dedicated to honing/sharpening your skills. There are 3 difficulty problems: easy, medium, and hard.

[HackThisSite](#) - Practice your 1337 hacking skills.

[ideone](#) - Web-based IDE with support for dozens of the most popular software languages.

[Cloud9](#) - Web IDE for Javascript, Python and Ruby with GitHub and BitBucket integration.

Web Development:

[250+ CSS Resources](#) - An enormous list of resources for mastering CSS

[CSS3 Generator](#)

[Smashing Magazine](#) - Blog about web (and some mobile) development from a design and user experience-based approach.

[Six Revisions](#) - Blog for web web development and design with some design tutorials for those unfamiliar with design.

Networking:

[Learn to Subnet](#)

Linux / *nix:

[Linux Documentation](#) - RTFM?

[DistroWatch](#) - A consolidated list of *nix distros.

[UNIX in 10 Minutes](#) - A good reference if you need to learn CLI fast.

Game Development: