# 10 Steps to Becoming a Professional Software Engineer

Written & Compiled for iD Tech Camps by
Audrey Van Norman & Christian Snodgrass

# 10 Steps to Becoming a Professional Software Engineer

> "There is great demand for skilled Computer Scientists."
>
> Mehran Sahami
> Stanford Associate Professor of Computer Science and Associate Chair for Education

Prepare to get excited about Computer Science – it's a dynamic field with endless opportunities for creative expression. And now is the perfect time to get involved.

Think about the millions of iPhone® and Android™ apps that people use every day. Think about Facebook apps, web apps and desktop apps. Think about almost anything you do on the computer – and you'll be thinking of the direct result of Computer Science.

Careers in Computer Science give you the opportunity to problem-solve, work on real-world issues and design programs that make peoples' lives easier every day. It's an incredibly rewarding field, and it's growing steadily.

Don't just take our word for it. As STEM (Science, Technology, Engineering, and Mathematics) initiatives have picked up steam in the educational world, industry demand for talented Computer Scientists has also increased. Mehran Sahami, a Stanford University Associate Professor of Computer Science and Associate Chair for Education, with a corporate background at industry giants Google and Epiphany, explains that now is the perfect time to pursue training in Computer Science:

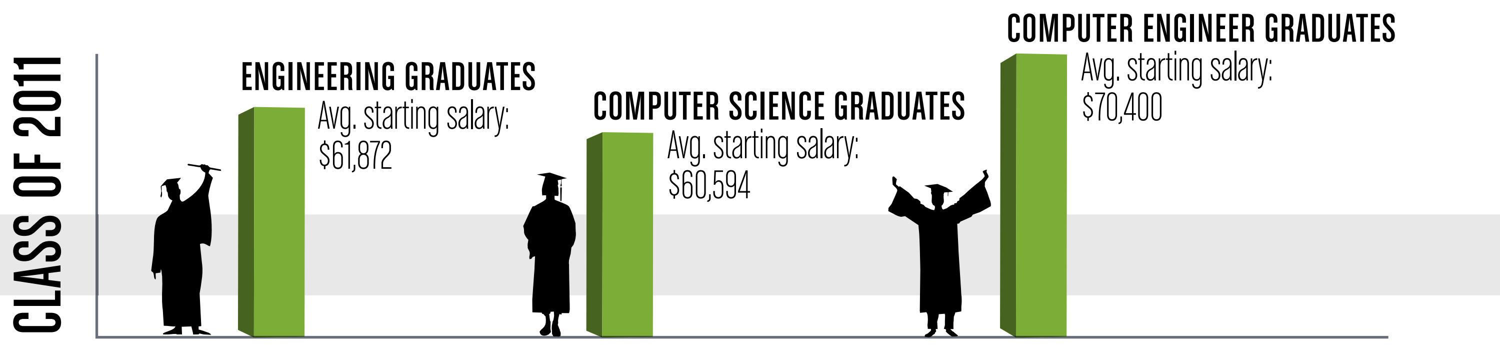
> *"There is a great demand for skilled computer scientists, with a wide variety of careers paths to choose from. Computing pervades so many aspects of our lives that it is becoming more and more essential for everyone to have some facility with computing. Computer Science is a field that marries analytical thinking, problem solving, and creativity in ways that help address some of our most pressing problems. Indeed, in the future, computing will continue to impact almost all aspects of our lives."*
>
> *Mehran Sahami*
> *Stanford Associate Professor of Computer Science and Associate Chair for Education*

**Entering the field of Computer Science can mean many things – you can eventually become a Programmer, a Software Engineer, or specialize in any number of subcategories such as mobile app development.**

## Ready to create your own iPhone® apps or develop Android™ apps?

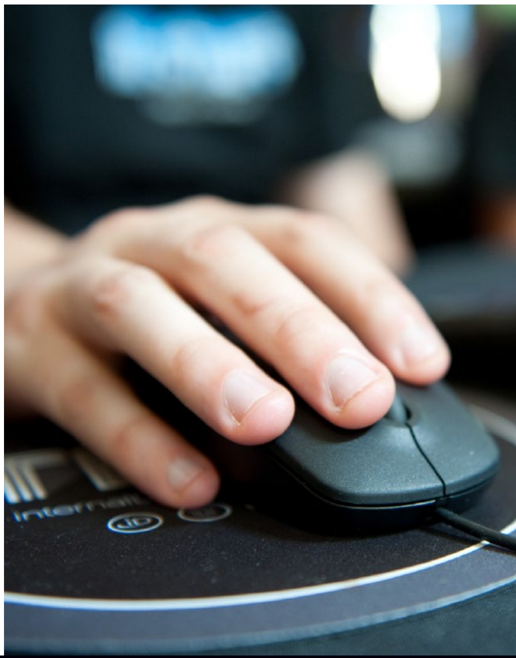
And if you need a little more convincing, here are a few interesting figures about the job market and associated beginning salaries. CNN Money published a study in January of 2012 about the previous year's college graduates. . .

*"The top earners of the Class of 2011 were engineering students, who were raking in average starting salaries of $61,872 upon graduation—a 1.5% increase from the previous year. Computer engineering majors were the highest-paid of that bunch, bringing in a whopping $70,400 a year. . . .Computer Science graduates weren't far behind, with the biggest increase in pay out of all the disciplines. Students in this field landed jobs with an average starting salary of $60,594 last year, up 4.1% from 2010."*



## Read on to learn about the path to becoming a professional Software Engineer

1. What is a Software Engineer?
2. Don't give up. It's hard. And rewarding.
3. Start early. Take classes. Turn coding into your 6th sense.
4. Develop core skills.
5. Experiment. Experiment. Experiment.
6. Build a portfolio with your experiments.
7. Experiment some more. Seriously.
8. Work with pros who are already Software Engineers.
9. Contribute to open source software.
10. Attend a school with a strong Computer Science program.



## Step 01:
What is a software engineer?

It should go without saying, but you need to know what you're getting into. What is the difference between a Software Engineer and a Programmer? The job titles can mean different things at different companies and are sometimes used interchangeably, but Software Engineers generally work on high level planning and Programmers work on implementation. A Software Engineer is the architect to a Programmer's construction worker. Both are invaluable, and many professionals wear both hats, sometimes simultaneously, over the course of a career. Expect to do a lot of programming on your way to becoming a Software Engineer.

### A typical day in the life of a Software Engineer can include:

- Generating high level project ideas
- Creating a spec document and planning out projects
- Coordinating with other programmers
- Programming (a.k.a. coding)
- Testing your programs
- Fixing bugs

### What can you expect once you become a Software Engineer? A rewarding career with tons of perks:

- Job security
- Great pay
- Flexibility
- Continuous growth and challenges

The U.S. Bureau of Labor Statistics projects employment in this field to increase by 32% from 2008-2018, much faster than the average 7-13% for all occupations. This rapid industry growth is fueled by continually evolving technologies and increasing software needs in the business world. Software Engineer salaries are also impressive – 2008 data from the Bureau of Labor Statistics showed that the middle 50% earned between $73,200 and $113,960. The highest 10% earned more than $135,780. There are many different fields to choose from once you know your programming – you can become a game developer, work on desktop applications, build websites, or even become a consultant or freelancer.

## Step 02:
Don't give up.
It's hard. And rewarding.

Bumps in the road are par for the course. The great thing about programming is that there is enormous diversity from project to project. You continually develop new skills and should never get bored.

You can never go on autopilot. Problem solving and creativity are absolutely vital. If you are a curious person, keep reading.

Don't become a Software Engineer because you want to cruise through the day. Become a Software Engineer because you love challenges and because obstacles only fuel your creativity.

## Step 03:
Start early. Take classes.
Turn coding into your 6th sense.

The best Software Engineers live and breathe code. As with any foreign language, the younger you start learning, the faster it sinks in and the better it sticks. In today's increasingly tech-savvy world, there are many options for students who want to learn programming.

One way to get a thorough introduction to programming is through a specialized class. iD Tech Camps and iD Programming Academy offer a series of programming courses for students as young as age seven. We've found that students who return year after year and build on their skills each summer attain the best results.

### Popular programming courses include:

- Adventures in Programming
- Programming in C++
- Programming in Java
- Programming – iPhone® & iPad® Apps in Xcode®
- iD Programming Labs 101 (iD Programming Academy)
- iPhone® and iPad® App Development – Objective-C & Xcode® (iD Programming Academy)
- Robotics Engineering & Coding (iD Programming Academy)
- Google Android™ App Development with Java (iD Programming Academy)
- Java Programming for the AP® Exam (iD Programming Academy)
- Programming for the Xbox 360® (iD Gaming Academy)

Of course, don't limit yourself. Having a firm foundation in many fields of technology – everything from game design to 3D modeling, Flash® animation and graphic design – will add to your abilities as a Software Engineer. A person who knows not only how to make it work – but how to make the navigation clean, the design crisp, and the user experience fluid – is the kind of person who will succeed.

## Step 04:
Develop core skills.

Know your stuff. Choose a basic language and get to know it inside out. After you've taken a course or done your own research, use your programming skills as much as possible to build a working foundation for yourself. Once you understand the fundamentals, you should be able to pick up new languages easily and choose whatever is best suited to a specific project.

### Some common languages that Software Engineers and Programmers use:

| | | |
|---|---|---|
| - C++ | - PHP/MySQL | - Microsoft .NET |
| - Java | - HTML/CSS/Javascript | - Ruby on Rails |
| - Visual Basic | - Flash/ActionScript | - Microsoft XNA |

## Step 05: Experiment. Experiment. Experiment.

Build a simple website. Recreate a game. Program anything, and don't be afraid to start small. You're gaining experience at this stage

One of the best things about iD Tech Camps and iD Programming Academy courses are their project-based curriculum. Students work to complete a project by the end of each week – awesome things like iPhone® apps. And these projects are a great way to get your foot in the app development door. Andrew took Programming (C++, Java) twice as an iD Tech Camps student, and then took iPhone App Development at iD Programming Academy. He left camp with three iPhone apps in progress, and continued experimenting after his session ended – now he has six apps in the Apple® store with 30,000 downloads between them. Not bad for experiments, right?

**"Andrew took Programming (C++, Java) twice as an iD Tech Camps student, and then took iPhone App Development at iD Programming Academy. Now he has 6 apps in the Apple® store with 30,000 downloads between them."**

We've surveyed some sharp programmers and engineers, and here is some advice for the budding programmer: set aside an hour every day to program. By the end of a week or two, you should be able to build a basic app. If you can't commit that much time, set aside a couple of hours every weekend. You can build up an app in about a month. The bigger and better the app, the more time it takes. . .but it's worth the effort. The important thing is to be specific about your concept (the thing you want to accomplish or the problem you want to solve) and stick to a schedule. This way you can make regular progress. Being self-motivated and sticking to a timetable are important.

## Step 06: Build a portfolio with your experiments.

Don't keep your experiments to yourself. Put them out there. Make your websites live. Publish your games. Share with the world. You want to be able to point future employers to real examples of your abilities.

Best of all, when you share your projects, you get hundreds of extra eyes on your work. Someone else might catch a bug that you didn't see, or anticipate a problem that didn't occur to you.

## Step 07: Experiment some more. Seriously.

Notice a pattern here? Try creating something else. Build a simple calculator. Make a desktop timer. Recreate classic arcade games like Pong. Build a task list manager. Create a simple MP3 player. Make a dice-rolling program.

And never leave well-enough alone. It can always be better. Go back to your old projects and make them prettier, make them faster, make them more functional. Experimenting and using programming skills in a practical, hands-on environment is the most valuable experience that any aspiring Software Engineer can have.

## Step 08:
### Work with pros who are already Software Engineers.

If you can intern for a great programmer, do it. Put in the time for training. It's the practice-makes-perfect principle, with an experienced programmer watching your back to make sure that you practice in the right way. You will learn more watching and working with them than reading ten thousand instruction manuals.

### Some big-name companies that offer internships:

- Microsoft
- Google
- Apple
- EA
- Ubisoft
- IBM

More and more companies are looking for up-and-coming talent. Get your foot in the door by interning at a reputable company. The long term payout will be worth it. They get to see you in action, and if you love what you do, it will show.

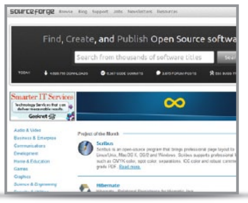
"More and more companies are looking for up-and-coming talent. Get your foot in the door by interning at a reputable company. The long term payout will be worth it."

## Step 09:
### Contribute to open source software.

As your skills improve and you gain confidence, branch out and work on open source software. You'll be rubbing shoulders with the best.

### Look at these sites for projects that interest you:

- www.gamedev.net
- www.sourceforge.net
- www.codeplex.com
- source.android.com

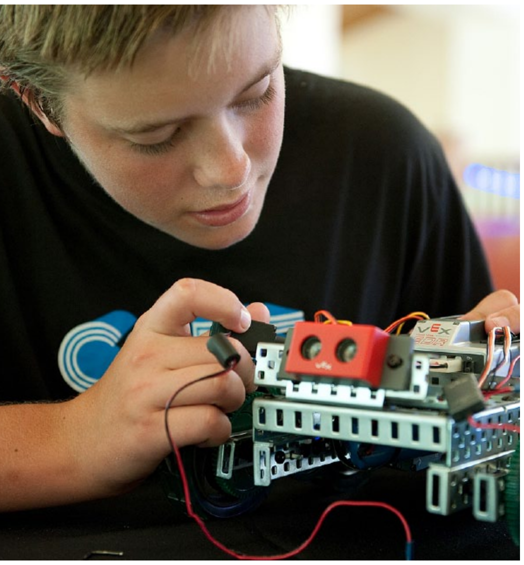


## Step 10:
Attend a school with a strong Computer Science program.

One of the big pluses of attending a great program is interacting with like-minded peers. When you rub shoulders with other programming enthusiasts at programs like iD Tech Camps and iD Programming Academy, this benefit becomes really clear. Peers in a well-organized program feed off of each other and the energy becomes contagious. If you can live and learn in that sort of environment, you will find inspiration and support at every corner.

### Some universities with prestigious computer science programs (that also host iD Tech Camps and iD Programming sessions):

- Massachusetts Institute of Technology (MIT)
- Stanford
- Carnegie Mellon
- UC Berkeley
- Princeton
- University of Wisconsin-Madison
- Harvard
- Brown
- UCLA
- Columbia University



A great way to get to know the feel of each campus is to attend summer sessions there – either iD programs, or in some cases youth education programs sponsored by each school.

After researching each school's degree programs, also consider location. Proximity to industrial and research parks can be a huge advantage. Many Computer Science departments have partnerships with large companies, and you may be able to get an internship or associate job during summers. Also, search around and sample some curriculum before you commit to a program – many schools (such as MIT) put all or part of their lectures online.

A word of caution: A degree is only worth as much as you put into it. Some students graduate with a grasp of the basic concepts of programming; some graduate with thousands of hours of actual programming experience. Employers will take someone with real-world experience over someone with fifteen different certifications. Make sure that you balance theory with practice.

**So does becoming a Software Engineer sound tough? It should, but don't let that phase you. Computer Science is an incredibly rewarding career path. Use these steps as a guideline and stay focused. Make yourself a schedule of classes and projects, and stick to it. Experiment, then experiment with your experiments. This is the beginning of a lifelong learning journey.**

iD Tech Camps®
www.internalDrive.com