GETTING YOUR FIRST SOFTWARE ENGINEERING INTERNSHIP/JOB
InternMakers was started by two passionate individuals who love to share knowledge they have acquired throughout the years to land internships and grow professionally. They would love for you to join this community and share on the knowledge with people who may need it.

You can follow InternMakers (@internmakers) on YouTube, Instagram, Facebook, and LinkedIn.

I really recommend you to follow InternMakers! I started this channel with Mariana Briones, who had two internships with Microsoft and will return as a full-time employee after graduation. The best way to get all the content (and support us) is by subscribing to the YouTube channel and/or following us on the other social media platforms.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVERVIEW</td>
<td>3</td>
</tr>
<tr>
<td>INTERNSHIP OPPORTUNITIES</td>
<td>4</td>
</tr>
<tr>
<td>Amazon</td>
<td>5</td>
</tr>
<tr>
<td>Dropbox</td>
<td>6</td>
</tr>
<tr>
<td>Facebook</td>
<td>7</td>
</tr>
<tr>
<td>Google</td>
<td>8</td>
</tr>
<tr>
<td>Microsoft</td>
<td>9</td>
</tr>
<tr>
<td>Palantir</td>
<td>10</td>
</tr>
<tr>
<td>Pinterest</td>
<td>11</td>
</tr>
<tr>
<td>Twitter</td>
<td>12</td>
</tr>
<tr>
<td>More Internships</td>
<td>13</td>
</tr>
<tr>
<td>Programming</td>
<td>14</td>
</tr>
<tr>
<td>Learning Java</td>
<td>16</td>
</tr>
<tr>
<td>Learning Python</td>
<td>18</td>
</tr>
<tr>
<td>Side Projects</td>
<td>20</td>
</tr>
<tr>
<td>Resume</td>
<td>22</td>
</tr>
<tr>
<td>Conferences and Career Fairs</td>
<td>23</td>
</tr>
<tr>
<td>Internship Framework</td>
<td>31</td>
</tr>
<tr>
<td>Intro to Interviews</td>
<td>32</td>
</tr>
<tr>
<td>Interview Preparation</td>
<td>33</td>
</tr>
<tr>
<td>Interview Tips by Google</td>
<td>33</td>
</tr>
<tr>
<td>Helpful Articles</td>
<td>35</td>
</tr>
<tr>
<td>Behavioral Questions</td>
<td>38</td>
</tr>
<tr>
<td>Preparation Map</td>
<td>42</td>
</tr>
<tr>
<td>HackerRank</td>
<td>44</td>
</tr>
<tr>
<td>LeetCode</td>
<td>48</td>
</tr>
<tr>
<td>Applying</td>
<td>50</td>
</tr>
<tr>
<td>References</td>
<td>51</td>
</tr>
</tbody>
</table>

© 2019 Alejandro Davila - All Rights Reserved
In this guide I will share a few things I wish I knew when I was a freshman in college. I will talk to you about opportunities that are available every year that go unnoticed by unaware or underprepared students.

I will set you up with a plan to prepare yourself for these opportunities and help you get that first software engineering internship.

Content:

- Internship Opportunities
- Java Programming
- Side Projects
- Resume
- Company Research
- Interviews
Many top tech companies recruit passionate students so that, in the future, these students can return to the company for more challenging internships or for full-time positions.

Internships are a great way to test potential employees and see whether they deserve a spot in the company when they graduate from school. They are also a great way for students to try out a job and a company to see if the role is something they’d like to pursue and if the company is a place they’d like to work.

These internships focus on passion for programming and uniqueness. Some of them request short essays to answer personal questions.

The following list contains a few of the summer internships. I am sure there are more opportunities available, but I will focus on these ones:
1. Amazon Future Engineer

**Description:**
The Amazon Future Engineer (AFE) internship program offers hands-on software development experience to college freshmen and sophomores who are making an early commitment to computer science.

Our AFE interns will partner with experienced software development engineers, who both help and challenge them as they work on projects that matter to our customers. They will be immersed in the exciting and innovative culture and make lasting connections with other AFE interns and Amazon pioneers.

**Deadline:**
December 31st

---

Taken from Amazon Career Site
Accessed on: December 2017
2. Dropbox Launch

**Description:**
The Dropbox Launch Program is a technical internship geared towards students who are looking for their first internship and a kick-start to their professional career - no prior experience required! Dropbox Launch is open to students of all backgrounds, and we particularly encourage applicants from historically underrepresented groups in technology.

Dropbox Launch aims to provide participants with a 12 week immersive experience at our San Francisco Headquarters. Our goal is to provide participants with an incredible and practical learning opportunity and help students apply university teachings to industry engineering. And if that wasn’t enough, Dropbox Launch interns receive a learning budget to further their accomplishments over the summer.

**Deadline:**
February

Taken from Dropbox Career Site
Accessed on: April 2018
3. Facebook U

**Description:**
Facebook University (FBU) is a paid eight-week training program designed to provide mobile development experience to students who are historically underrepresented in computer science.

The eight-weeks is broken down by three-weeks of mobile development training in either iOS or Android, followed by five-weeks of hands-on project experience in a small team setting. Throughout the duration of the program you’ll be paired with an engineering mentor who will help guide you along the way.

**Deadline:**
January 15th

Taken from Facebook Career Site
Accessed on: December 2017
4. Google Engineering Practicum

**Description:**
The Engineering Practicum (EP) program is a 12-week developmental opportunity for first and second-year undergraduate students with a passion for technology – especially students from historically underrepresented groups including women, Native American, Black, Latino, Veteran and students with disabilities.

Our unique internship offers the opportunity to work on a software project alongside other EP interns and full-time Googlers, and gives you the chance to bridge the gap between academic understanding and practical professional experience.

**Deadline:**
November

---

Taken from Google Career Site
Accessed on: December 2017
5. Microsoft Explore

Description:
Explore Microsoft is a 12-week summer internship program specifically designed for college underclassmen (freshmen and sophomores) and offers a rotational experience that enables you to gain experience in our different software engineering roles.

This program is designed to give you hands-on experience with various tool and programming languages in the field of software development, and encourage you to pursue degrees in computer science, computer engineering, or related technical disciplines. Your on-the-job learning will be augmented with mentoring, community building and networking opportunities.

Deadline:
November

Taken from Microsoft Career Site
Accessed on: December 2017
6. Palantir Path

**Description:**
At Palantir, we’re passionate about creating opportunities for all individuals to grow and succeed. We launched Palantir Path in 2017 to support students who are new to the tech world, especially those from *underrepresented communities* who may not have had exposure to technical opportunities before college.

As a Palantir Path intern, you’ll become part of the Palantir family – and challenged accordingly with high-impact work. You’ll be in a Software Engineer role as part of a team of extraordinary engineering talent supporting a specific product or Palantir customer on multi-faceted, open-ended, and complicated problems. You’ll be paired with a mentor who will help you learn to build real-world software and to think critically about how problems can – and can’t – be solved with technology.

**Deadline:**
October

Taken from Palantir Career Site
Accessed on: September 2018
7. Pinterest Engage

**Description:**
The Pinterest Engage Scholars Program is dedicated to providing growth and learning opportunities to *Freshmen* and *Sophomore* students enrolled in a bachelor’s degree program with a demonstrated interest in majoring in a technical major. We believe that building diverse teams is essential to building great products and encourage applications from *underrepresented* groups in the field of computer science. This program series is an *unpaid training opportunity* over the course of *two months* through June and July and is located in San Francisco, CA.

**Deadline:**
April 16th

---

*Taken from Pinterest Career Site*

*Accessed on: April 2018*
8. Twitter Academy

Description:
Twitter Academy is a paid 12-week summer internship program designed for second-year students interested in gaining hands-on and real-life experience in the tech industry. This program is for students historically underrepresented in computer science (black, Latinx, Native American, and/or female.)

Deadline:
Mid November
9. More Internships

Most companies offer internships for Juniors or Seniors, so it is hard to post all of them here. What I can do is give you a list of companies so that you can go to their website and look for opportunities.

Click here to find internships. Here you will find a list of companies and a guideline on how to look for available positions.

If you are looking to work at a startup, go to AngelList.
If you are looking for a public company, go to Fortune 500 companies.
If you are looking for a private company, go to Unicorn’s List.
If you are looking for software companies, take a look at this list.
If you are completely new to programming it is important to have a study plan. All programming languages are different somehow, so you must become proficient in at least one of them so that it becomes easier to learn the rest. These are some of the areas you can focus on:

**Web Development**

*HTML and CSS*: if you want to build website you will have to learn these two. HTML is the skeleton of the website, and CSS is the styling of it. [Click here to learn HTML.](#)

*JavaScript*: the programming language of the web browser, it turns static websites into dynamic and cool websites. [Click here to learn JavaScript.](#)

*Web Frameworks*: these frameworks are used to have more robust and structured web applications: React (JavaScript), Angular (JavaScript), Ruby on Rails, Django (Python), jQuery (JavaScript)
Software Development

**Java**: widely used object-oriented programming language. Mostly used to develop Server Applications, Web Applications, Mobile Applications (Android), Desktop Applications, Enterprise Applications, and Games

**Python**: powerful scripting language. It is maybe one of the easiest to learn because of its simple syntax. Mostly used for Web Development, Data Science (machine learning, data analysis, data visualization), and Scripting (automate tasks).

**C#**: object-oriented programming language developed by Microsoft. It is used in most of Microsoft’s products. Mostly used to develop Server Applications, Web Applications, Mobile Applications (Android and iOS), Desktop Applications, Enterprise Applications and Games

**C++**: object-oriented programming language. It is good for performance and efficiency. Mostly used for Web Applications, Desktop Applications, Enterprise Applications, and Games

**C**: low-level programming language. Can connect hardware and software. Mostly used for Operating Systems, Compilers, Assemblers, Network Drivers, Data Bases, Utilities, and more.

This guide will focus on Java and Python for now.
This section is mainly for high school students, freshmen students, or people with no prior programming experience.

There are countless tutorials and courses online that teach Java for beginners. I will list a few of them (all free) that should help you start programming in Java.

Most of these websites will require you to register an account to save your progress.

Free Java courses for beginners: (complete one of these ones to go to the next section)

- Java Programming Basics - Timeline: approx. 6 weeks (Udacity)
- Java Tutorial for Complete Beginners - 16 hours of material (Udemy)
- Intro to Programming in Java - The complete CS1 Course by Princeton University
- Teach Yourself JAVA in 21 Days - Free online book

One important website I will focus on is HackerRank. For this section you should at least complete all the introductory level Java and Algorithm challenges.

- Introduction Challenges - Timeline: approx. 3 days or less (HackerRank)
Once you are familiar with Java, the next step is to dive deeper and learn about Algorithms and Data Structures. Another good subject to learn is Object Oriented Programming.

The following courses are recommended. Algorithms, Part I covers elementary data structures, sorting, and searching algorithms. Algorithms, Part II focuses on graph and string processing algorithms. Object Oriented Programming covers the fundamentals to work on programming projects using a GUI (Graphical User Interface). Data Structures covers fundamental dynamic data structures, including linear lists, queues, trees, and other linked structures; arrays, strings, and hash tables.

Free Java courses for intermediate level:

- **Algorithms, Part I** - Timeline: approx. 6 weeks (Coursera)
- **Algorithms, Part II** - Timeline: approx. 6 weeks (Coursera)
- **Object Oriented Programming** - Timeline: approx. 6 weeks (Coursera)
- **Data Structures** - CS course from Berkeley

Complete the first three courses and the last one for extra practice and you will be better prepared for interviews.
This section is mainly for high school students, freshmen students, or people with no prior programming experience.

Free Python tutorials for beginners: (complete one of these ones to go to the next level)

- Dive Into Python - Online book
- Learn Python - Interactive Python Tutorial

For this section you should at least complete all the introductory level Python and Algorithm challenges from HackerRank.

- Introduction Challenges - Timeline: approx. 3 days or less (HackerRank)
Once you are familiar with Python, the next step is to dive deeper and learn about Algorithms and Data Structures.

Free Python tutorials for intermediate level:
- Problem Solving with Algorithms and Data Structures - Interactive Python Tutorial
- Data Structures - Interactive Python Tutorial

Complete at least one of those courses and you will be better prepared for interviews.
<table>
<thead>
<tr>
<th>Importance of Side Projects</th>
<th>Open up a deeper and engaging conversation.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Demonstrate your ability to build, to apply what you have learned overall, and to actually code.</td>
</tr>
<tr>
<td></td>
<td>Demonstrate your ability to learn on your own and dabble with new tools and technologies.</td>
</tr>
<tr>
<td></td>
<td>Demonstrate drive, passion, self-motivation, and the ability to work on unstructured problems.</td>
</tr>
</tbody>
</table>

**Click here for a list of 120+ project ideas**
If you don’t know where to start, a good first project can be building a **personal website**.

- **Creating Websites**
  - *Intro to HTML/CSS*
  - *Making webpages interactive*

Other extracurricular activities that employers like to see:

- **Hackathons**
- **Open Source Projects**
- **Undergraduate research**

[Click here for more information about how to get started with extracurricular activities](#)
Resume screeners will focus on **how smart** you are and **how well** you can code. For this reason your resume should **highlight** those two things.

Here are some suggested guidelines for writing your resume:

**Length**
- Keep resume to one page

**Projects**
- Include 2-4 significant projects

**Employment History**
- Include only relevant job positions
- Discuss accomplishments, not job descriptions: “Show what you did, how you did it, and what the results were”

**Software Languages and Technologies**
- Don’t list Microsoft Office
- List languages with experience level:
  - Java (expert), C++ (proficient),
  - JavaScript (prior experience)

[Click here for more tips and sample resumes.](#)
CONFERENCES AND CAREER FAIRS

Why should you attend a conference?

- Educational, technical, and career opportunities for students
- Workshops (leadership, technical, professional, and more)
- Networking
- Career Fairs!
CONFERENCES AND CAREER FAIRS

STEM/Computing Conferences

Grace Hopper Conference
September, 2018

HENAAC Conference
October, 2018

SWE Conference
October, 2018

SHPE Conference
November, 2018

© 2019 Alejandro Davila - All Rights Reserved
CONFERENCES AND CAREER FAIRS

Workshops and Networking

- Download the conference app
- Look at the agenda
- Make a plan
- Attend the workshops and talks that interest you the most
- Attend Hospitality Suites (rooms reserved by specific companies)
- Talk to other attendees, ask questions during sessions, and expand your network!
Upload your resume early to the conference website (important)

Get a list of the companies attending

Have at least 15 copies of your resume

Master your elevator pitch

Dress professionally

Carry a portfolio, if possible

Be prepared!
CONFERENCES AND CAREER FAIRS

Elevator Pitch = Tell Me About Yourself

- Name
- Major, Classification, University
- Pre-Professional Experience (Past and Present) - Skills Gained
- Future Job/Internship/Research
- Personal Interests (Optional)
- Hook the recruiter!

© 2019 Alejandro Davila - All Rights Reserved
CONFERENCES AND CAREER FAIRS

Elevator Pitch Template

“Hi, my name is __________ and I’m a __________ (year in school) at __________ (college or university) studying __________ (major) and __________ (minor, if applicable). I am interested in pursuing work in the __________ industry/field and have participated in __________ (extracurriculars, organization, volunteering). I have developed these skills __________ by __________ (participating in an internship, working a particular job, etc.) “
Go through every company’s website and look for positions that align to your interests
Be attentive to positions that are specifically open during the conference period
Rank the companies by your level of interest
Write the booth number for each company
Write a short description of your top ranked companies.
(Optional) Make a map and filter your companies by booth location
## Career Fair - Questions for Recruiters

Fill out this table as a reference guide when selecting company booths to visit.

Print this table along with the questions for interviewers (pg 36).

Study your printed page before talking to recruiters.

<table>
<thead>
<tr>
<th>My rank</th>
<th>Fortune 500 Ranking (optional)</th>
<th>Booth #</th>
<th>Company Name</th>
<th>Description</th>
<th>Why this company?</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>12</td>
<td>132</td>
<td>B Company</td>
<td>Description B</td>
<td>I love B Company because Y</td>
</tr>
</tbody>
</table>

© 2019 Alejandro Davila - All Rights Reserved | 30
Before applying to a company, you should know what the company is looking for in candidates.

I designed this table to help you compare companies and find out which skills are required for every position.

**Action:** find at least 10 companies/positions and add them to the table.

<table>
<thead>
<tr>
<th>Company</th>
<th>Position</th>
<th>Required/Preferred Skills</th>
<th>Application Deadline</th>
<th>Application URL</th>
<th>Recruiter/Contact</th>
<th>Status</th>
<th>Applied on</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>X Comp</td>
<td>Software Engineering Intern</td>
<td>Java, C++, Python, JavaScript, or other.</td>
<td>November 15</td>
<td>xcomp.com</td>
<td><a href="mailto:recruiter@xcomp.com">recruiter@xcomp.com</a></td>
<td>Applied Accepted Rejected More...</td>
<td>MM-DD-YYYY</td>
<td>Follow up in two weeks</td>
</tr>
</tbody>
</table>

**Glassdoor** is a great resource to find information about a company (culture, employee reviews, salaries, interview questions and interviewee reviews). Filter by position to look for specific information.

[Click here for a list of companies] [Click here to access a collaborative table]
It is now easier than ever for companies to filter out applicants interested in programming related roles. This is the usual interview process:

Coding challenges have become more common, and it is time for students to prepare for them. These should be taken seriously.

Technical interviews are very similar to coding challenges; it is critical to practice.

Most important subjects

- Data Structures
- Algorithms
- Big O
Substantiate
Make sure that you substantiate what your CV/resume says – for instance, if you list Java or Python as your key programming language, questions about this are fair game and may be asked of you.

Explain
Explain your thought process and decision making throughout the interview. In all of Google's interviews, our engineers are evaluating not only your technical abilities but also how you approach problems and how you try to solve them. Many of the questions asked in Google interviews are open-ended because our engineers are looking to see how you break down and approach the problem. There is no “one true answer” to them. We want to understand how you think. This would include explicitly stating and checking any assumptions you make in your problem solving to ensure they are reasonable. Think out loud.

Clarify
Ask clarifying questions if you do not understand the problem or need more information. Many of the questions asked in Google interviews are deliberately underspecified because our engineers are looking to see how you engage the problem. In particular, they are looking to see which areas you see as the most important piece of the technological puzzle you’ve been presented. Clarifying questions are encouraged!
Also, take interviewer tips seriously because they’re trying to give you helpful hints. They may ask some additional questions or share comments when you're solving a problem. Take a minute to think through these prompts -- they can help you land on a more optimal solution!
Improve
Think about ways to improve the solution. Share the different options or tradeoffs you’re considering. Be flexible. In many cases, the first answer that springs to mind may need some refining. It is worthwhile to talk about your initial thoughts to a question. Here are some guidelines:

- You want to get to the simplest solution as quickly as possible. You can start with a brute force solution, but then...
- Discuss tradeoffs. How would you improve your solution? How do you make it faster? Use less memory? Know the time and space tradeoffs of the solution/data structures you pick.
- Code. We generally don’t want pseudocode. It might be acceptable in some limited cases. If you are not sure, ask the interviewer. The interviewer will guide you through.
- If you don’t remember the exact interface to a library class or method, that is OK, as long as you let the interviewer know, and you postulate a substitute with a conforming interface.

Practice
Make sure you practice writing code on paper or a whiteboard in preparation for in-person interviews, or code in a Google Doc for phone interviews. Be sure to test your own code and ensure it’s easily readable without bugs. Keep in mind, you can choose one of the following programming language you’re most comfortable coding in: C, C++, Java, Javascript & Python

Ask Questions
At the end of the interview, most interviewers will ask you if you have any questions about the company, work environment, their experience, etc. It’s always good to have some pre-prepared for each interview.
Extra Prep Resources

**Online resources**
- Dean Jackson’s [ACM article](#)
- [Cracking the Coding Interview Video](#)
- [Five Essential Phone Screen Questions by Steve Yegge](#)
- [Google+ Technical Interview Coaching Session](#)
- [TopCoder Tutorials](#)

**For Underclassmen**
- [Guide for Technical Development](#)
- [How to Get Hired - What CS students need to know](#)

**Books**
- [Programming Interviews Exposed: Secrets to Landing Your Next Job](#)
- [Programming Pearls](#)
- [Introduction to Algorithms](#)
- [Programming Interviews Exposed; Secrets to Landing your Next Job](#)

**Get some practice!**
To practice for your interview, you may want to try:
- [Project Euler](#)
- [ACM-ICPC Live Archive](#)
- [UVa Online Judge](#)
- [UVA Toolkit](#)
- [Programming Challenges](#) (Text with solutions)
**Articles for Interview Prep**

Get That Job at Google
Get That Job at Facebook
How to Prepare for Technical Interviews
ABC: Always Be Coding
Four Steps to Google, Without a Degree
How to Get a Job at Google [part 1] [part 2]
Preparing for a technical interview with programming contests
How to Prepare for Tech Interviews
How To Win the Coding Interview
How To Pass a Programming Interview

**Books**

Gayle McDowell - *The Google Resume*
Gayle McDowell - *Cracking the Coding Interview*
Giguere, Mongan, Kindler - *Programming Interviews Exposed*
Aziz, Lee, Prakash - *Elements of Programming Interviews [code]*
Narashima Karumanchi - *Coding Interview Questions*
Cormen, Leiserson, Rivest, Stein - *Introduction to Algorithms*
## Miscellaneous Articles

<table>
<thead>
<tr>
<th>Name</th>
<th>Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 8 Mistakes in Technical Interviews According to Data</td>
<td>Interview Tips</td>
</tr>
<tr>
<td>The Guerrilla Guide to Interviewing (v.3)</td>
<td>Interview Tips</td>
</tr>
<tr>
<td>Done, and Gets Things Smart</td>
<td>Career Advice</td>
</tr>
<tr>
<td>Five Essential Phone Interview Questions</td>
<td>Interview Tips</td>
</tr>
<tr>
<td>50+ Interviews with Facebook, Twitter, Amazon &amp; others</td>
<td>Personal Experiences with Interviews</td>
</tr>
<tr>
<td>Technical Interviews Make Me Cry</td>
<td>Interview Tips</td>
</tr>
<tr>
<td>How I hustled my way to a developer job at Khan Academy</td>
<td>Personal Experience and Career Advice</td>
</tr>
<tr>
<td>Career Advice</td>
<td>Career Advice</td>
</tr>
<tr>
<td>We Only Hire the Trendiest</td>
<td>Career Advice</td>
</tr>
<tr>
<td>How I went from failing every interview to a job at Amazon</td>
<td>Interview Tips</td>
</tr>
<tr>
<td>‘Clean your desk’: My Amazon interview experience</td>
<td>Personal Experience with Interviews</td>
</tr>
<tr>
<td>How I Hire Programmers</td>
<td>Interview Tips</td>
</tr>
<tr>
<td>Google’s &quot;Director of Engineering&quot; Hiring Test</td>
<td>Google Tips</td>
</tr>
<tr>
<td>Resume helper</td>
<td>Resume Tips</td>
</tr>
</tbody>
</table>

© 2019 Alejandro Davila - All Rights Reserved
Behavioral questions help companies determine if the candidate’s personality, motivations, and past experience are relevant to the role and company’s culture.

They are important questions and should not be ignored. Even the most skilled programmers have been rejected because of poor preparation for behavioral questions.

Click here for a complete list of interview questions

<table>
<thead>
<tr>
<th>Common Behavioral Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tell me about yourself.</td>
</tr>
<tr>
<td>Why do you want to work for X Company?</td>
</tr>
<tr>
<td>Talk me through a time you were working in a team and someone wasn’t pulling their weight.</td>
</tr>
<tr>
<td>Can you talk about a challenge you faced in the past? How did you overcome it?</td>
</tr>
</tbody>
</table>
Go through your resume and make sure you **know everything** about each **project/job** listed in your resume. Use this grid to come up with different personal **stories**. You should know these stories **by heart** at the time of your interview.

<table>
<thead>
<tr>
<th>Common Topics</th>
<th>Job 1</th>
<th>Job 2</th>
<th>Project 1</th>
<th>Project 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Challenges</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mistakes/Failures</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Experiences Enjoyed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Leadership</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conflicts</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>What You’d Do Differently</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
INTERVIEW PREPARATION

How do you know the interview will be behavioral?

- “15 minute chats”
- First round interviews (most situations)
- Interactions with recruiters (Career Fairs)

Tips

- Go to Glassdoor.com and look for real interview questions from X Company
- Navigate through X Company’s website
- Read about them!
- Look for current news and come up with questions
- Study the preparation grid
### Questions for interviewer (always ask questions!)

#### Questions about position
- What are the top skills that you look for in candidates applying to this position?
- How does the company measure performance for the interns? What are the company's systems for feedback?
- What kind of projects have other interns worked on during this internship?

#### Questions about company
- What is the company culture like?
- What do you enjoy most about working for the company?
- Are there opportunities for ongoing training through your organization?
- What percentage of interns come back as full-time employees?

#### Question to conclude
- May I contact you with further questions? Do you have a business card?
Before Interview

1+ Years
- Build projects outside of school/work
- Learn multiple programming languages
- Build website/portfolio showcasing your experience
- Find a programming job and take classes with large projects

3 - 12 Months
- Work on more projects
- Create a draft of resume and get it reviewed
- Make a list of target companies (Company Research section)
- Read the Interview Preparation section of this guide
- Learn Big O and basic data structures and algorithms

1 - 3 Months
- Practice interview questions
- Do several mock interviews
- Practice more interview questions

4 Weeks
- Create interview preparation grid
- Review/update resume
- Begin applying to companies
- Read this guide again
- Do more mock interviews
Preparation Map

Before Interview

1 Week
- Do a final mock interview
- Rehearse stories from interview preparation grid
- Master Big O, basic algorithms and data structures
- Practice more interview questions

Day Before
- Rehearse stories from interview preparation grid
- Practice more interview questions and review mistakes
- Print notes for phone screen

Day Of
- Wake up early to eat a good breakfast and be on time
- Be confident! (not cocky)
- Talk out loud to show your mind process
- It’s okay to be nervous. Stumbling and struggling is normal!

After
- Write thank you note to recruiter
- If you haven’t heard back from recruiter, check in after one week.
- If no offer, ask when you can re-apply. Don’t give up!
- Get an offer? Celebrate! Your hard work paid off!

Inspired by Cracking the Coding Interview 5th Ed.
As I mentioned before, I will focus on HackerRank to practice for the interviews. There are many other tools and resources one can use to practice, but the benefit of practicing with this one is that many companies use it for coding challenges.

Work on the following tracks:

**TUTORIALS**
- 30 Days of Code
- Cracking the Coding Interview

**CORE CS**
- Algorithms
- Data Structures

**LANGUAGES**
- Java
INTERVIEW PREPARATION

HackerRank

30 DAYS OF CODE

Improve your coding skills by coding for 30 days in a row

- Unlock a new code challenge and tutorial each day
- Submit solutions in Java, C++, and other popular languages
- Learn if-else statements, recursion, data structures, object-oriented programming and more

hackerrank.com/domains/tutorials/30-days-of-code

Tutorial videos by Kathryn Hodge
INTERVIEW PREPARATION

Interview Preparation Kit

Curated challenges and tips based on learnings from 1000+ companies to help you prepare for your upcoming interviews.

View

hackerrank.com/interview/interview-preparation-kit
Difficulty levels in HackerRank are: **Easy**, **Medium**, and **Hard**. Start with the Easy challenges and move your way up until you finished every challenge from that level.

<table>
<thead>
<tr>
<th>Track</th>
<th>Mandatory Subdomains</th>
<th>Optional Subdomains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>• Warmup</td>
<td>• Dynamic Programming</td>
</tr>
<tr>
<td></td>
<td>• Strings</td>
<td>• Programming</td>
</tr>
<tr>
<td></td>
<td>• Sorting</td>
<td>• Graph Theory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Search</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Recursion</td>
</tr>
<tr>
<td>Data Structures</td>
<td>• Arrays</td>
<td>• Trie</td>
</tr>
<tr>
<td></td>
<td>• Linked Lists</td>
<td>• Heap</td>
</tr>
<tr>
<td></td>
<td>• Stacks</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td>• Introduction</td>
<td>• Object Oriented Programming</td>
</tr>
<tr>
<td></td>
<td>• Strings</td>
<td>• Programming</td>
</tr>
<tr>
<td></td>
<td>• Data Structures</td>
<td>• Graph Theory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Search</td>
</tr>
</tbody>
</table>
LeetCode is another great tool to practice and prepare for technical interviews. The great thing about LeetCode is that people can add questions, solutions, and even vote on questions that have been asked in actual interviews.

1) To get started, create an account.
2) Click on Problems in the top menu
3) Click on Top Interview Questions
4) Choose the difficulty (start in Easy)
Try to complete as many Easy problems as you can before trying to tackle the Intermediate ones. You will see that the Easy problems are not as simple as they seem!

Tip: look at the solutions only after you tried to solve the problem on your own. If a problem has no official solution, you can look at the discussions for that problem and see what other users came up with as solutions.

You can use the Tags to find problems for specific topics:
You are now a step closer to becoming a Software Engineer!

It is important for you to look for opportunities because they will not look for you.

Use the company list you created and apply to every position applicable to you.

Tailor your resume to each position.

Apply, apply, apply!

This guide was developed with support from CAHSI. CAHSI is a nationwide alliance that works to recruit, retain, and advance Hispanic students in computing fields.

More information: http://cahsi.cs.utep.edu/
These are all the sources that helped me develop this guide:

− Gayle Laakmann Mcdowell - Cracking the Coding Interview 5th Edition
− Google - Preparing for Google Technical Internship Interviews
− HackerRank
− LeetCode
− Amazon Career Site
− Dropbox Career Site
− Facebook Career Site
− Google Career Site
− Microsoft Career Site
− Palantir Career Site
− Pinterest Career Site
− Twitter Career Site
− Glassdoor
− Taishi-Y - Interview-ToDo
− Michael Deng - How to land a top-notch tech job—while you’re still in school