



Yeshiva University

www.yu.edu

YESHIVA COLLEGE

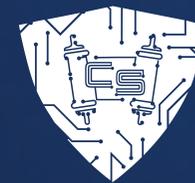
Computer Science Internship & Job Search Guide

For any questions or comments
about this guide, please contact

Judah Diament

Department Chair and Professor,
Computer Science, Yeshiva College
Belfer Hall, Room 1110

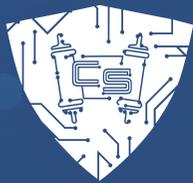
diament@yu.edu



YESHIVA COLLEGE
COMPUTER SCIENCE

WELCOME TO THE YESHIVA COLLEGE

Computer Science Internship & Job Search Guide



YESHIVA COLLEGE
COMPUTER SCIENCE

Contents

INTRODUCTION.....	2
TIMELINE.....	4
YEAR 1.....	4
YEAR 2.....	7
YEAR 3.....	8
YEAR 4.....	10
CRITICAL JOB SEARCH SKILLS AND ACTIVITIES	13
WRITING YOUR RESUME.....	15
Format.....	15
Content.....	15
Cover Letter	17
Have Your Resume Reviewed.....	17
INTERVIEWING.....	18
Master the Game.....	18
Interview Types and Process.....	18
Preparation for Technical Interviews.....	19
Preparation for Behavioral Interviews.....	19
Interview Prep Groups with Other Students and/or Alumni.....	21
NETWORKING.....	23
LinkedIn	23
Referrals.....	25
Company Programs for Students	25
Campus Recruiting Events	25
APPLYING FOR JOBS	26
Places to Search for Job Postings.....	26
Apply to Many Jobs	26
General Tips.....	26
NOTES.....	28

Introduction



WELCOME TO THE YESHIVA COLLEGE COMPUTER SCIENCE (YCCS) INTERNSHIP and Job Search Guide. This guide gives you an idea of what you need to do (beyond your classwork), and when you need to do it, in order to increase your odds of graduating with a great job in hand. We do not cover every possible permutation of events that you may encounter—each person’s process and experience can differ in any number of ways—and there is no way to guarantee that you will receive a full-time offer. However, not following this guide radically increases the odds that you will not have a job upon graduation. You should check in with this guide at least once a semester to ensure you are on track to graduate with a full-time job in hand.

Ideally, your Senior year will be free of job search stress because you will have accepted a full-time offer from the company you interned with during the previous Summer.

This is a common approach in the Software Engineering hiring process: companies use Summer internships to select the students to which they will offer full time jobs. The offers come at the end of the Summer, and jobs begin after graduation. We say “ideally”

because it takes a very significant amount of time and effort to receive an internship offer and, when piled on top of your coursework, it can feel somewhat overwhelming. You may be hesitant to put the time in, or you may make it a low enough priority that you just never “have time” for it. If graduating with a great job in hand is a significant factor in why you go to college to begin with, doesn’t it make all the sense in the world to dedicate the time which is required for a realistic internship/job search?

There is one critical part of getting a great C.S. job not covered in this guide—learning C.S.! Our C.S. curriculum has been designed to push you hard towards reaching your potential as a Computer Scientist. Putting your best effort into all aspects of your C.S. education is the most important contributor to the long-term health of your C.S. career. This guide is intended to cover the things you need to beyond your C.S. courses to get a great C.S. job. In addition, keep in mind that a number of your C.S. professors have decades of industry experience, and you should look to them for guidance on career matters, not just technical C.S. matters.

There are two sections to this guide: **Timeline**, which outlines what job search activities you should be engaged in during each semester of your four years on campus, and **Critical Job Search Skills and Activities**, which gives concise guides to the various aspects of the job search. You must read, integrate, and internalize both sections to know the appropriate steps to take towards securing internships and, eventually, a full time job.

For any questions or comments about this guide, please contact Professor Judah Diament at diament@yu.edu.

**WE WISH YOU THE BEST
AND HOPE YOU FIND
THIS GUIDE HELPFUL!**



Timeline | Year 1

SUMMER BEFORE

- This summer is not critical, so feel free to do something totally unrelated to career building. This is a great time for learning, chess, etc.
- At the same time, non-technical work experience is still work experience, and as such it can help you get a first technical internship the next Summer. If you want to get a jump on things, take whatever interesting job you can get.

FALL

- Keep in mind that from a C.S. perspective you are a freshman, even if you have 32 credits from Israel!
- Create a [LinkedIn](#) profile. You can find guidance regarding what your profile should contain at bit.ly/3mGxWn7. Remember that you must remain 100% professional on LinkedIn—this is not Facebook or Instagram!
- Spend some time looking at internships, paying attention to the small number of them that are open to first year students. (After your first year, you'll be a "Rising Sophomore".)
- Networking:
 - Get to know the upperclassmen, even the ones who are not your TAs. They will be great resources down the road for career advice.
 - Attend any/all networking events or trips to/with tech companies.
 - Build relationships with contacts at those companies.
 - Remember to connect on LinkedIn afterwards with anyone you meet.
 - LinkedIn
 - Follow companies which you are interested in and begin connecting with employees at those companies.
 - Include messages with your connection requests. (See the networking section on pages 23–25.)

SPRING

- Create a public [GitHub.com](#) profile. The source code of any impressive CS projects that you complete should be published there, with each project getting its own repository in your profile. Your public GitHub is where you should showcase your best CS work, so potential employers who Google you can see the great stuff you've been doing:
 - The GitHub repositories you use for submitting your homework are not public, so if you want to publish a school project, you will have to copy it over from your YU repository to a public repository you own.
- Choosing a Track in YCCS:
 - If you aren't sure what track you would like to take, you can delay your choice until next semester by taking both Linear Algebra and Calculus II this semester.
- Unless you get lucky, you will most likely not find a Software Engineering Internship for the Summer. If you find a non-technical internship, discuss with a mentor before accepting.
- Assuming you don't find an internship, you have a few choices:
 - Sign up for the YC CS Summer course which will teach you a lot about real-world software engineering and will look impressive on your resume. This is by far the best choice for the vast majority of students.
 - Alternatively, you can work on a large project of your own over the summer, which you can point to as a sample of your work and talk about in interviews.
 - Lastly, some smaller companies don't host an internship program, but if you reach out to them and offer to be an unpaid Software Engineering intern, they may be willing to give you the experience. Finding such an opportunity involves networking to find a company and reaching out to individuals at the company, be it via personal connections or via [LinkedIn](#).

Timeline | Year 2



SUMMER BEFORE

Follow through on the Summer plans made in the Spring, and don't forget to take a couple weeks off to relax and recharge.

FALL

- Spend time every week looking for, and applying to, internships. Apply to >75 companies.
- Begin practicing for technical interviews by doing coding challenges on HackerRank/Leetcode:
 - The premium versions may be worth looking at.
 - Aim for 1–2 a week.
- By the end of this semester, you need to have selected a track in the CS major.

SPRING

- If you don't find an internship, sign up for the YC CS Summer course or work on a large independent project over the summer.
- Continue practicing HackerRank/Leetcode. Aim for >2 a week.





SUMMER BEFORE

- Starting in June, watch for job postings at financial companies that are targeted at new college graduates. Many of these applications open during the Summer and close very early in the Fall semester.
- Continue practicing HackerRank/Leetcode. Aim for >3 a week.

FALL

- Spend a significant amount of your time looking for and applying to Internships:
 - ⇒ The Summer after your third year is your last internship Summer and it will have a large impact on your options upon graduation.
 - ⇒ If you are serious about finding a good internship for this summer, apply to >100 places.
- Continue practicing HackerRank/Leetcode:
 - ⇒ Most of your time should be taken up by finding, applying to, and interviewing.
 - ⇒ You should be very comfortable with taking on most medium HackerRank/Leetcode challenges.



SPRING

- The applications for most internship at large companies have closed at this point, but opportunities can sometimes still be found at smaller companies.
- If you do not have any offers, network to find any software engineering internship, even if it is unpaid at a startup or non-profit organization, in which you will tackle technical challenges at the appropriate level and demonstrate that you can successfully function in a real-world organization.
- If you do not have any prospects for this summer, you should make plans to do some advanced software engineering project over the summer. The project should be something you could imagine creating a startup around, or open-sourcing. The YC CS Summer program is not advanced enough for what you need for your resume at this point.
- Think about why you do not have an offer for the summer:
 - ⇒ Did you get interviews/coding assessments? If not, have a well-informed person in industry review your resume and assess what first impression you are giving.
 - ⇒ Did you fail coding assessments? If so, do more studying on CS fundamentals (Data Structures, Algorithms, Math for C.S.).
 - ⇒ In some situations, it is just luck of the draw. You know how much effort you put into the search.

Timeline | Year 4

SUMMER BEFORE

- This internship is your audition for a full-time role with the company. If you perform well, at the end of the Summer they will likely make you an offer for a full-time role upon graduation:
 - ☞ Work hard and perform well at your internship.
 - ☞ Show them that you can work independently and be a productive member of their team.
 - ☞ If you need help, ask for it, but try to solve technical challenges independently—don't immediately ask your boss or teammates for help.
 - ☞ If there are company/bureaucracy issues, you can ask the right person (your boss, a teammate, HR, an internal mentor, etc.) about those without too much hesitancy—you are not expected to know anything about those types of things. At the same time, don't be a pest!

FALL

- Hopefully you will have a full-time offer from the company you worked at in the Summer.
- If you don't have an offer or are not happy with your offer:
 - ☞ Proceed as you did the previous Fall: apply to > 100 jobs, practice HackerRank/Leetcode.
 - ☞ Network very aggressively to find, and/or prepare for, opportunities.
 - ☞ Look at application deadlines and requirements for graduate school (MS in CS), take the GRE, and apply. Even if you would prefer not to go to graduate school, you don't want to graduate from college with no plan for what comes next. If you apply to graduate school, get accepted and enroll, and then get a good job offer, you can always back out of graduate school.

SPRING

Continue your search and application processes.





Critical Job Search Skills and Activities

Following are guides to various skills and activities that are essential parts of the job search. Make sure to put real and persistent effort into each one of these areas.

Writing Your Resume

Your resume must paint a picture of who you are and what you know. Recruiters (the HR people who review and filter resumes of people applying for a job) take an average of **7 seconds** to read your resume, and that is how much time your resume has to convey a clear and consistent message that makes you stand out from other applicants.

In addition to recruiters, your resume will often be read by technical people who will interview you for a job. Depending on the company, those interviewers may be members of the team which you applied to join. As such, in addition to hitting the right notes for the recruiters, your resume has to be a good starting point for an in-depth conversation about your C.S. experience.

FORMAT

- At this stage in your career, your resume should fit on a single page.
- If you run out of space, play with the formatting but ensure margins do not get smaller than half an inch all around. Resumes are occasionally printed and it's unprofessional for your resume to be cut off by the printer.
- Text should be broken up into bullet points; there should be no long blocks of text in your resume.

CONTENT

1. Identify yourself

- Name, Email Address, Phone Number, link to your LinkedIn profile, link to your public GitHub page (assuming you have a public GitHub account and there are public projects there).

2. Education

- School, degree/major/minor, expected graduation year and month.
- GPA (unless it is so low enough that it will do more harm than good).
- If you are in the honors program, on dean's list, etc., include that there.
- Relevant coursework: list off CS and Math courses.

3. Experience

- Any CS work experience you have goes first.
- Earlier in your college career this is likely to include more school and/or personal projects. Hopefully Summers will give you the opportunity to get some work experience on there as well. For each school and personal project:
 - Provide a link to a GitHub repository which contains its code. You should have a separate Github Repository for each project.
 - Each project must have a well-written readme.md file.
 - List technologies used.
 - Very briefly describe key CS challenges in the project and how you solved them.
- Don't list things like "camp counselor" unless you have nothing else to list and/or it's something really impressive, e.g. you were a counselor at HASC and cared for an extremely disabled camper, etc.

4. Languages, Frameworks, and Tools

- These include Java, C, C++, Python AND Node.js, Junit, Maven, GitHub, etc. Any language or framework listed should be something you are comfortable answering questions about. Be sure to research what the common interview questions are for each language/framework/tool you list.

5. Extracurriculars & Volunteer Work

- Are you on the board of any clubs?
- Do you participate in any relevant club activities?
- Are you active in the YU ACM Chapter?
- Are you a Peer Tutor? RA?
- Do you volunteer anywhere, e.g. Tomchei Shabbos? Any other activities that indicate you are a selfless and caring human being?

COVER LETTER

A cover letter is mostly worthless for any good tech job; focus on crafting the resume itself. Below is a quote from someone in Human Resources at one of the top tech companies in the world. The name of the company has been replaced with "*****".

*When applying to ***** , the cover letter adds almost no value –and that largely applies to any major tech company too, simply because the cover letter is designed to fill in the gaps of a resume that functions as merely the first step in the application process. ("The resume's not going to get you the job, it's just going to get you the interview.") Especially when applicant volume is high, the resume is far more valuable than a cover letter. A situation where a cover letter *is* useful is when it provides information to the recruiter that a resume can't; e.g. an explanation of a gap in employment, background information about why a GPA is lower than an applicant would like it to be, etc. But overall I think 90% of people don't need a cover letter to apply to *****.*

HAVE YOUR RESUME REVIEWED

Many students simply never received the right resume-building guidance/advice and have very poorly written resumes—this can lead to automatic rejections. Your first step is to ask upperclassmen who have gotten good internships/full time jobs, to review your resume. Once the "big mistakes" have been filtered out, ask alumni who work in companies you'd like to apply to if they can give you resume feedback. Remember that they are busy people, so ask them weeks ahead of any deadlines!

**REMEMBER THAT THEY
ARE BUSY PEOPLE, SO ASK
THEM WEEKS AHEAD OF
ANY DEADLINES!**

Interviewing

MASTER THE GAME

The interview process for software engineering jobs is recognized by insiders to be very imprecise and idiosyncratic. Most knowledgeable people would tell you (in moments of honesty) that there are great candidates who get rejected, and not-so-great candidates who somehow get hired. While mastery of C.S. is your key to long-term success, you also must become an expert in the “interview game” despite the fact that there is often very little overlap between the day-to-day work of a software engineer and the challenges (technical and behavioral) of the interview process. Because of this disconnect, winning the interview game requires a significant amount of preparation and effort beyond the C.S. curriculum!

INTERVIEW TYPES AND PROCESS

There are two main types of interviews: technical and behavioral. Both of these types of interviews can come in various forms—an online assessment on a web site (whether it’s answering questions or writing code), a phone call, a video interview, or an interview “onsite”, i.e. at the company.

Most companies will send an assessment within a couple days of receiving your application (banks tend to send HireVues, tech companies tend to send coding assessments.) Some companies will “ghost” you and never send you an assessment. Ghosting is not personal - these companies just don’t have the bandwidth to respond to everyone. (To avoid being ghosted, make sure your resume is a good fit; if you are ghosted, think about why they may not have prioritized you as a candidate and update your resume accordingly.) If they do send you an assessment, do it ASAP; a slow response to the assessment indicates less enthusiasm on your part, and also gives others time to get in line before you and take the job before you even interview. “You snooze, you lose!”



PREPARATION FOR TECHNICAL INTERVIEWS

Below is a list of activities and resources you should use to prepare for technical interviews.

LeetCode/HackerRank

- ⇒ YC CS requires students in the two algorithms courses to take weekly coding challenges. This is a fantastic introduction to the process, but you MUST practice on your own time as well.
- ⇒ It is worth your while to purchase the premium version of these services, which include full mock assessments, solutions with runtimes better than your own, etc.

Books

- ⇒ *Cracking the Coding Interview: 189 Programming Questions and Solutions*
 - This is a must-read, with great strategies for approaching a wide range of questions.
 - In recent years, interviewers have used this book when creating interviews.
- ⇒ *Programming Interviews Exposed: Secrets to Landing Your Next Job*
- ⇒ *The Algorithm Design Manual*
- ⇒ *Elements of Programming Interviews in Java: The Insider's Guide*

PREPARATION FOR BEHAVIORAL INTERVIEWS

- Have a clear personal story about your interest and experience in C.S. to communicate to an interviewer.
- Do many practice interviews—it takes a lot of practice to get good at it.
- Research the company. Recruiters like to see that you:
 - ⇒ Know all about the company, including the products and services it sells.
 - ⇒ Have a good reason why you want to work for the company.

- Research the Interviewer. They have varied backgrounds, so if you are able to research them in order to tailor your approach in the interview to their background, do so!
- Be able to speak about your resume:
 - ➔ Practice speaking about projects with both technical and non-technical people.
 - ➔ Relate past experiences to the qualifications for the role.
 - ➔ Work on your 60 second response to “*tell me about yourself*” Many interviewers begin with this!
- Standard Questions to prepare for: “*Tell me about a time when you...*”
 - ➔ ... were challenged
 - ➔ ... problem solved
 - ➔ ... failed, and what you learned from it
 - ➔ ... took on a leadership role; how did you do so and what did it entail
- General Tips:
 - ➔ Give complete and thoughtful answers, but don't drone on and on.
 - ➔ Humor is risky—it can help, but can sink you if taken the wrong way. When in doubt, avoid it.
 - ➔ Be 100% honest, but think before volunteering information.
 - ➔ Send a non-generic thank you note after every interview to each person involved:
 - “I particularly enjoyed discussing x and learning about y with you.”
 - Keep it relatively short—this is a short but thoughtful “Thank you”, not a position paper!

INTERVIEW PREP GROUPS WITH OTHER STUDENTS AND/OR ALUMNI

- Interviewing is not easy—almost everyone needs practice.
- A great way to prepare properly is to form a small group of students from your YC CS cohort who will conduct mock interviews with each other to polish your interview skills collectively.
 - ➔ Behavioral interview skills to practice:
 - You should learn to talk about everything on your resume, in addition to respond to the questions listed above.
 - Sit still—don't swivel or squirm in your chair.
 - Look the interviewer in the eye.
 - ➔ Technical:
 - Learn to explain what you are doing and why you are doing it while solving technical problems.
 - Learn to both take hints as well as accept feedback from the interviewer regarding your approach, without getting offended, losing confidence, or getting defensive.
 - **Whiteboarding** is a high stakes interview, it usually means this is the final round. Practice for it as you will need to get it right the first time!
- Reach out to upperclassmen and alumni who you know to ask for advice and mock interviews.



Networking

Talking (not emailing—talking, as in communicating with another human being either face to face or over the phone/zoom) to people currently working in your target industry accomplishes two very important things. First, it gives you a reality-based perspective on the company or role that the person works at/in and helps build and calibrate your understanding of the industry in general. Second, many companies will give preferential treatment (via one mechanism or another) to candidates that have someone inside the company who is willing to recommend (a.k.a. “refer”) them. Below are a number of resources / activities you should take advantage of to build your professional network.

LINKED IN

LinkedIn is the most important tool/site for networking. You should create a LinkedIn profile for yourself that has many of the same qualities as your resume (in terms of painting a picture of you), and then set out to connect to people in roles and companies that you’d like to one day be in.

A list of all alumni from YU who are on LinkedIn can be found at bit.ly/3mzq4nf (a LinkedIn account is required to access this resource). When you’re applying for an internship or full-time position, filter this list by said company to find alumni who work at the company, and connect with them on LinkedIn. For LinkedIn instructions, go to bit.ly/3CCsqan. You should include a message when trying to connect that introduces you to the person you are trying to connect and explains why you are connecting, e.g.:

“Hi \$NAME, my name is \$YourName and I’m a Computer Science student at Yeshiva College. I’m planning on applying for an internship at \$Company for this Summer. Do you have some time to speak? I’d love to hear more about what you do at the company and any advice you can share on how to stand out. Thank you!”



The likelihood that a connection will help you increases significantly if you are able to speak with them. Though email/LinkedIn may feel more comfortable, a personal connection will go a long way. When speaking to a connection, first ask questions about him/her. “What is your role and what do you work on?”. “What keeps you at that company?” After asking several questions about themselves, you can begin to ask for their advice regarding the hiring process, how to make yourself most likely to be a desirable candidate, etc.

It is critical that you keep in mind that they are doing you a big favor, and as such you should be thankful to them and respect their time. Among other things, respecting their time means coming prepared with good specific questions and if they say, for example, that they have to go, don't try to keep them on the phone any longer. Additionally, if the call was scheduled to last 30 minutes (for example), do not keep talking for more than 30 minutes without asking them if they have to go or if they can spare more time.

REFERRALS

Many companies have an internal referral system. If they do, getting a current employee to refer you can provide a huge boost to your application. Specific points to be aware of include:

- Some companies require referrals be submitted before you submit your application. Research this about each company you are applying to and act accordingly.
- Contacts who know you well will often be willing to submit a referral, so don't be shy about asking.
- Asking an alum/na to refer you can help—many are willing to help you by virtue of you being a YU student.
 - If an alum does not offer, you can ask if they would be comfortable referring you.
 - Ensure you give them a way out when requesting—be understanding if they would prefer to not refer someone who they do not know well.

COMPANY PROGRAMS FOR STUDENTS

There are several large companies which provide networking events and materials for interns, we suggest you sign up to receive their emails and invites. One example is **Connect With Google**. Do your research and find out if a company you are interested has such a program, sign up and take advantage of it if they do.

CAMPUS RECRUITING EVENTS

The Yeshiva University Career Center runs some recruiting events.

Tips:

- **Depth is better than breadth:** recruiters meet a lot of people, meeting all recruiters at an event briefly will produce fewer results than meeting a few recruiters and creating a good impression or forming a relationship with those few.
- **Follow-up:** if you can get contact information for the recruiters you spoke with, send a thank you email and ask for advice on next steps. Again, recruiters meet many people and it is those who stand out that they will speak with the most.

Applying for Jobs

You must be aware of when during the year your target companies recruit for Summer and full-time jobs and apply as early in their process as possible. Applying later generally decreases your chances.

PLACES TO SEARCH FOR JOB POSTINGS

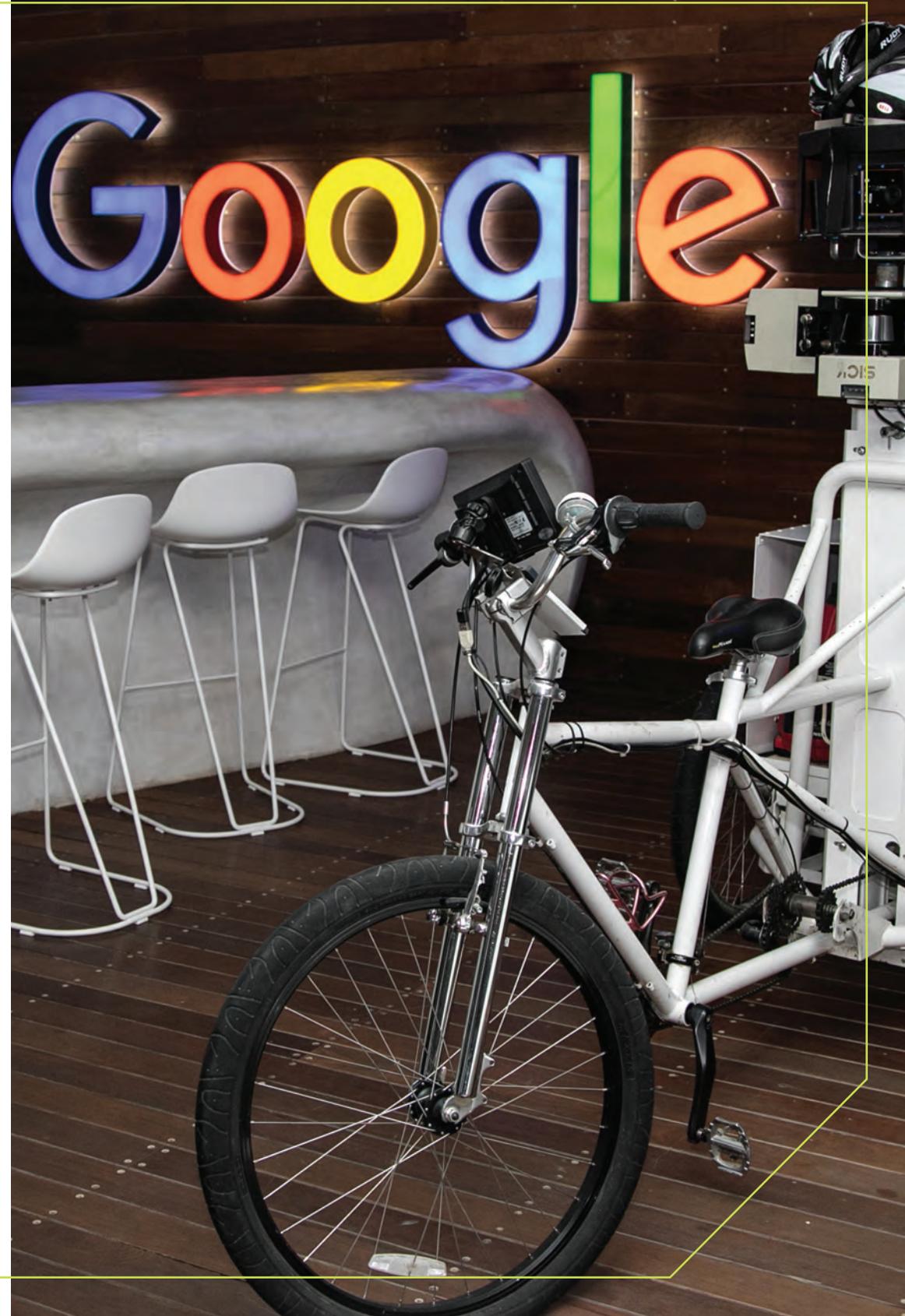
- Websites of the companies for which you'd like to work
- [LinkedIn.com](https://www.linkedin.com)
- [StackOverflow.com](https://stackoverflow.com)
- [Indeed.com](https://www.indeed.com)
- www.levels.fyi/still-hiring/
- Student collated lists, such as github.com/pittcsc/Summer2022-Internships

APPLY TO MANY JOBS

You have no idea what companies will give you an interview, let alone make you an offer. You should apply to many (>100) companies and jobs, not just your top 25, to have realistic odds of getting opportunities you are happy with. In addition, every interview at any company is very important practice, even if it is at a company you would not accept an offer from.

GENERAL TIPS

1. Do not stop applying until you have accepted an offer.
2. Designate a specific, dedicated time every single week for applying to jobs. This is a time-consuming task, and it will get "starved out" of your schedule if you don't designate inviolable time for it.
3. Track your applications. When applying to so many different companies and contacting different connections, it is easy to lose track of your application status and phone calls you should be making. There are various tools you can use to keep track of it all. Spreadsheets are a great tool to keep track of this information, but you may notice that you've outgrown a spreadsheet if you are applying to enough places and speaking to enough people. Take a look at **[Kanban-Board-Style](#)** software that may help in tracking your application statuses. YU provides access to **[Microsoft Planner](#)** which implements this style or application.



Notes