getting-tech-internship (or new grad job)

This document is written and reviewed by Carleton students (class of 2022 and 2023) as a guide for Carleton students on finding tech internships (or new grad positions). Our contacts are listed at the end of the document.

Why a tech internship?

- ➔ **Work experiences** are very important in the tech world. Since anyone can pretty much make their own website, application, etc. anytime, job searching gets very competitive with skilled workers. Getting internships from well-known companies will give you a very huge edge over others. Internships from very well-known companies can significantly increase your chances of getting interviews for both internships and new grad positions.
- ➔ People may not realize that tech internships **pay A LOT** of money. Most well-known companies will be paying you at least $40/hr plus free corporate housing, transportation, relocation, and a lot more (health insurance, money for random stuff, food, etc.). In some cases, you can pretty much use your internship money to pay tuition at Carleton. Can check the list of salaries for internships out here: 2022 Tech Internship Guide & Salaries | Levels.fyi
- ➔ It's a lot of **fun**. Companies love their interns, so there are a lot of events with free food, awesome guests, and gifts. For example, Microsoft hosts their intern event (festival with big-name artists like Olivia Rodrigo, Pitbull, and Chainsmokers), intern game (a weekend driving around Washington solving puzzles), and intern puzzle every year. Pretty much every tech company also has Q&A events with their CEOs, CVPs and executives. Some inspiration to apply: What's the Microsoft in-person internship experience like?
- ➔ You **learn** a bunch of new technologies and best industry practices. It's super satisfying when your project goes to prod. You build connections and get to present your project to executives/higher-ups.

What do technical interviews look like?

Each company has different ways of interviewing its applicants. Their goal is to filter out candidates that they think will not be a good fit for their companies. As of 2022, companies like Meta don't give out Online Assessments, but they just give you technical and behavioral interviews if you pass their resume screening. Other companies like Google and Amazon give out Online Assessments and technical rounds. More information can be found in Cracking the Coding Interview book (very recommended, the Career Center has one).

Online Assessments (OA)

This is a portion where companies will give you coding questions for you to solve in a limited amount of time. It can be just one question for 60 minutes or 3 questions for 80 minutes. This really depends on the company. Its main goal is to filter out candidates that “cannot code” or do not meet the bar. This portion is meant to be very hard in terms of the difficulty of the questions and the fact that there is no human factor in the assessment. They might choose to ignore your application if your score on the OA is not good enough. The way to prepare for this is leetcode (more on this below). You may skip this step if you have a strong resume or other connection with the company (scholarship winner, previous offer, referral, etc.).

Some common OAs include Hackerrank, CodeSignal (common strategy: do questions 1, 2, 4 and if you have time open and do question 3; anything below a 750 is probably not good enough; should aim 800+), Byteboard (more real, design doc and then a small codebase that you have to add some functionality).

Resume Screening

This process may either happen before or after OA and may be the part that people with low experience struggle with the most. Note that this process can be done by either a bot or a recruiter (or both). A recruiter from Dropbox confirmed that they do not use bots to filter resumes while other companies like Google do use bots to filter applicants. Getting referrals is said to guarantee that at least your resume will be read by a human (though might be a myth). Again, each company is different and may have different things that they are looking for on your resume. Once you have a solid internship, you don’t need to worry about this step as much, passing the resume screening will be relatively easy. See strategy on resume below.

For this step, having a polished Linkedin profile can help, sometimes recruiters reach out directly via email/Linkedin dm and you don’t even need to worry about applying. Bonus: Create a Linkedin (if you have Amazon student prime, you can get 6 months free of premium).

Phone Screening

This is a part where a recruiter will call you and ask you some basic questions about yourself and about why you are interested in the company. Some companies, like Microsoft, will ask you some basic computer questions like “What is a thread?” Others might be asking you very simple programming questions or just basic behavioral questions like “Give an example of when you have conflicts with your coworkers and how you resolve the conflicts?” This part is meant to figure out your red flags and if you are interested in the company or not (Non-FAANG may ask “What about X interests you?”, “What do you know about X?”). Again, each company is different. While some can be very technical, some may not have this round at all.

Technical Round(s)

This is the most important round of the process. Most of the time, this will be a back-to-back interview, meaning that you will be getting 2, 3, or 5 consecutive interviews (around 1 hour

each) with different interviewers. You will be asked to talk a little bit about yourself before each interview, and you will be given at least one programming question (leetcode medium-hard).

Some companies like Meta and Amazon may start off the questions easy and keep asking you to optimize the solutions and/or ask more questions that are built on top of the previous ones. In the end, you will get a chance to ask questions to the interviewers, which is a chance for you to show your interest in the company or to present yourself in a way that fits with the personality that the company wants.

To practice, do leetcode (see more on this below) and practice thinking out loud. This part is mostly standardized throughout the industry, but there are also some companies that ask you to do something unique. For example, Amazon will expect you to show how your answers fit under their Leadership Principles. Palantir has 3 different kinds of interviews for this part. There are technical, which is like normal tech interviews; decomposition, which is kind of similar to system design interviews; and learning, where they will give you new computational concepts for you to learn on the spot and solve problems that they give you.

Hiring Manager Round

This round can either be mainly behavioral or technical. It should be similar to the previous technical rounds with more emphasis on behavioral questions and big picture thinking. If you reach this step, congratulations, it means that you are very close to an offer.

System Design interview

Some new grad and very few internship interviews have a system design round (Oracle OCI has a system design interview but Google explicitly asks not to include those questions to new grads). If you have some background, reading this system design cheatsheet is probably enough. But in case you need something more detailed, read the “System design interview” book by Alex Xu (there’s a copy in the Career Center), or this system design primer repo. Topics that may come up (entry level system design tend to more more simple and not focus on scalability):

- Load balancing
- Caching
- Multithreading and concurrency issues
- Networking
- Vertical and horizontal scaling
- Database replication and partitioning
- Asynchrony and queues

How should you prepare + strategies + timeline?

Each company has different things that they are looking for. Mainly, they will be assessing you on your personal experience and technical skills. Personal experience can be evaluated by your

resume and how you answer behavioral questions. Technical skills can be evaluated by how well you do on the technical portions of the interview. Big companies like Google, Amazon, Meta, Microsoft, etc might have a more standardized interview process while smaller companies or startups might have different ways to evaluate their interns. For example (real story), if you have a lot of experience building a startup by creating your own application, smaller companies can be careless about how well you do on technical interviews and give you an offer based on your experience while you will still need to pass technical interviews to get offers from big companies. On the other hand (also real story), if you only have personal projects and no working experience, you can still get offers from big companies given that you pass their resume screening and technical interviews while smaller companies or startups might not be spending as much time on you. Therefore, the best way to maximize your chances of getting an internship is to find companies that fit your personal experience and technical skills. However, there is no way to know which company fits you, so the strategy is to **apply to a lot of places.** It is very common to be applying to 50 or even 200 companies to eventually get an offer, especially if this is your first internship.

Build your Resume

Remember that your resume will probably be read by either a bot or a recruiter. Thus, the important things are to put good keywords (so the bot can pick up) and impact-focused wording (so the recruiter can understand). In general, your resume should make sense for people with non-tech backgrounds and should contain keywords that show your technical skills. For example, “Built a gamification ASP.NET web application that is expected to boost [the company]’s sales by 13% by effectively monitoring sales performance from 12 different countries”. This allows the bot to pick up that this person has experience in developing a web application, and a person (with no technical background) to understand the impact of the project. A strategy to improve your resume is to let a lot of people read it, especially those who are working for tech companies.

If this is your first time finding an internship, you should spend a lot of time making projects in your free time. Keep in mind that the laptop that you are currently using is more than capable of creating an application, a website, a game, and a lot more. With unlimited free resources online, what you need to do is to spend some time on it and create something that shows your interest in the field. Note that any work experience related to tech is very valuable. Your first and second internships might not be from well-known companies, but you should be able to get better and better internships or jobs as you gain more experience.

Without a good resume, no matter how good you are at technical interviews, you might not even receive a single interview if you don’t pass the resume screening. Read this resume guide and update your resume: How to Write a Kickass Resume and This is what a GOOD resume should look like. Don’t forget to get it reviewed.

Prepare for Technical Interview

Without good preparation for a technical interview, it would be impossible to pass one no matter how many interviews you receive.

The only way to practice is to do a lot of problems. More specifically, do problems by topic and try to learn the pattern of problems that they will be asking. Leetcode is a site that contains a lot of technical problems categorized by difficulty (easy, medium, and hard). Interviews for internships will mostly be medium or hard. The bare minimum that you should do is this list Blind 75 LeetCode Questions (video solutions), also take a look at the most common patterns. Easy problems on leetcode are mostly useless, and some of the hard problems are mostly too hard, so you should focus on medium problems. People who really want to be very prepared will aim for 100 - 200 problems. Note that each person has a different speed and style of learning so focus on what works best for you. Also, it really depends on the companies on how hard the technical interviews will be, but you can likely expect to get at least a medium leetcode question for most technical interviews.

This process takes a lot of time. For Carleton students looking for your first internship, you should start leetcoding 1-3 questions every day at the beginning of your summer (at the latest) and keep doing it until you get an offer.

After you have been leetcoding for a while, get together with your friends and try to do mock interviews: each picks a random question on leetcode or other sites and lets another person do it. While doing it, explain out loud your thought process and have another person assess your explanation and give you hints when you are stuck. Doing problems by yourself is very different from explaining your thought process out loud while thinking about it. For post-pandemic, whiteboard interviewing might come back: you write code on a whiteboard in front of your interviewers. Therefore, you will need a lot of practice.

For reference, here are some common data structures and algorithms tested (from a medium article):

Data structures

Preliminary (must know)	Basic (must know)	Intermediate (should know)	Advanced (rarely necessary)
array	linked list	priority queue	binary indexed tree
string	queue	disjoint set (union find)	segment tree
set	stack	min/max heap	Merkle tree
hash table	binary search tree	trie	

	graph	monotonic queue/stack	

Algorithms:

Preliminary (must know)	Basic (must know)	Intermediate (should know)	Advanced (rarely necessary)
basic programming constructs	recursion	dynamic programming	Knuth–Morris–Pratt (KMP) pattern matching
common operations on data structures	sorting (merge, quick, bubble, selection)	shortest paths (Dijkstra's, Bellman-Ford)	minimum spanning tree (Prim's, Kruskal's)
	array search (binary, linear)	topological sorting (DFS, Kahn's)	maximum network flow (Ford–Fulkerson)
	tree traversal; BFS and DFS		

Prepare for Behavioral interviews

First, prepare a short intro for the "tell me about yourself/introduce yourself" question that is not just a repetition of your resume, you are probably going to get that one in 50% of the interviews. For the other questions, use the STAR method - Situation, Task, Action and Result; prepare a few good technical stories before the interview. I suggest creating a table with some key theme, for instance:

Theme	Situation	Task	Action	Result
Failure and what I learned	It was week 6 of my internship and …	I had to develop a service that …	My code had a key bug, … so I ..	I reconsidered the design and … I learned that

For the themes, look at the company values, pdf the recruiter probably sent you, and past questions. Always prepare some questions for the end.

Referrals / Networking

Each company values referral differently. Some companies value it a lot, and some may just not care. However, it will unlikely hurt you if you actually have one. When asking for referrals, the

companies will be asking the referrer to write something about you. This means that you will have to know the person for them to write something that will help you get an interview. However, some referrer might be asking you to write that statement yourself: you pretend that you are the referrer and write a referring statement about yourself. Some companies like Meta and Microsoft will directly ask you to write a short essay. Note: the level of the person referring you to the job may be important (at Microsoft there's little consideration for referrals from new grads and interns, so you should ask for referrals from more experienced employees in this case).

It is also important to note that if you get referred by someone and you end up accepting the offer, the person that refers you will get a bonus. This means that the person that refers you in will have your back. A good way to ask for referrals (if you don't personally know the person) is to schedule a call with them to introduce yourself and ask about their experience in the company. At the end of the call, you can then ask them to give you a referral if they are comfortable with it.

Most companies will only allow you to be referred to positions that have already been posted on their website. This means that if you want to apply for a position as soon as possible (keep in mind that the spots for internships are limited), you will have to make sure that the person that is referring you is responsive enough for them to quickly give you a referral when the company opens their application (which is very unpredictable).

When networking with other people, try to learn about the companies that they are working for. If you talk to a lot of people from the same company, you will likely see a trend in what kind of people the company hires or whether you will enjoy working in the company or not. One way to think about this is, say if you are interviewing for Google and get asked "Why are you interested in Google?" How can you answer this question in a way that is unique to how thousands of other people answer the same question? Keep this in mind while networking to make the most of your time.

Cover Letter

Tech companies normally don't read cover letters due to the number of applications they received. If they want you to write something, they will require you to write that on their job application: i.e.: Why do you want to work with us? Tell us more about yourself. etc. It is probably more worthwhile to use your time writing cover letters to apply to 4 or 5 more jobs. Even companies that mention that a cover letter is required in the job description may not care if you actually submit one and can move you in the process (real example: NBCUniversal).

Timeline

Tech companies start recruiting very early. Many companies will start opening their applications as early as July (July 2022 for June 2023 internships). This will keep going until their spots are filled up, which depends on the companies. Given that each company can only take a limited number of interns, you should apply to these companies as soon as possible before the spots

start to fill up. Some of these companies will take very long (a couple of months) to get back to you while some will at least send you an OA right away, so the best way is to be prepared before July and keep preparing while you are applying.

Freshman/Sophomore

Summer/Fall

- Try out some leetcode but don't sweat too much. It's hard for a freshman or a sophomore to even get an interview, so practicing now might not be worthwhile. However, if you are serious about getting an internship or if you have enough experience to secure interviews, then practice a lot.
- Apply to all tech internships targeted for first and second-year students: Google STEP, Microsoft Explore, Uber Star, etc. More on CarlTech.
    - The interviews for these internships tend to be less technical. The questions are easier than the questions for a regular SWE internship, but there are much fewer spots (so apply early) and mostly reserved for sophomores.
    - These internships will help you a lot when you find a new internship next year, or you can do well in the internship and keep getting return offers until full-time. In a sense, you can pretty much get a full-time job offer if you manage to get one of these internships. Even if you don't get a return or don't want to return, your resume will be very strong, and you will get interviews for SWE internships pretty much anywhere.
    - If you are a returning intern, you will most likely get a sweet bonus and a raise and can maybe join the company full-time at a higher level (with higher pay) if your organization allows that (one level above a regular new grad).
- Apply to other SWE internships
    - This is a way to let you know firsthand how hard the interview process is.
    - If you do get an offer, that will really help you for other internships

Winter/Spring

- Start some projects and find opportunities to get experience. Something like being a CS course staff or working at a local company is a very good starting point. Take a look at part-time internships (16-24h/week): Warner Bros. Discovery (HBO Max, …), Fox News, Box, Wix, The Home Depot, HPE, Intel, NBC Universal, Wall Street Journal, Proofpoint, Crowdstrike, and IBM.
- Talk to alums and try to get referrals from them.

Junior/Senior

Summer

- Practice leetcode. It's probably too late to start a new project now (given that you already have something on your resume), so focus on nailing the technical interviews.
    - Start doing mock interviews with friends

- Companies start their application process as soon as mid-July, so keep your eyes on their website (there is no shortcut to check, you just have to go to their website and check every day).
- Summer 2023 internship list [GitHub - pittcsc/Summer2023-Internships: Collection of Summer 2023 tech internships!]

Fall

- You will start getting a lot of OAs and interviews. Keep practicing while you are interviewing, even if you feel very confident in some of the interviews you have already done (since it is very unpredictable whether you get an offer or not)
- While you are interviewing, you should keep applying to other companies. Don't be too set on any company since there are so many great companies to work for in this field.

Winter/Spring

- Keep applying and don't lose hope.

Got the offer? Congrats!

First, congrats! There are a couple of options:

- If you have offers and are satisfied with them, it is time to compare the options. [Choosing between companies | Tech Interview Handbook]
    - Reach out to employees from the company on Linkedin, and maybe read some posts in [Blind] (an anonymous professional network) or [r/csmajors] (however, those can be a little toxic so don't spend much time there).
    - When comparing compensation, take into consideration the other benefits. For instance, offer 1 maybe $50/hour in HCL and offer 2 $46/hour in MCL, but if you take into consideration that offer 1 offers $4500 (pre-tax) for housing and offer 2 $7000 (post-tax), offer 2 is better. Also, remember that compensation is not the most important thing, especially for internships.
- The offer you got is not your first choice.
    - Reach out to recruiters you are already in contact with and tell them that you got an offer and ask them to expedite the process. Only mention the company name if it's "impressive". This signals competence and you may be given an onsite.
- If you are a **new grad**, you may be able to **negotiate** (interns normally can not negotiate internship offers, but I have seen it happen at Apple and hedge funds). However, this is somewhat delicate so please do your research or talk with a career staff. It is highly likely it will work if you have competing offers from big tech companies/the company's direct competitors (Microsoft and Amazon or Google).
    - Resources: [negotiating-major-swe-offers] (for new grad returning interns), [Offer Negotiation] email template, [Ten Rules for Negotiating a Job Offer - haseeb qureshi]
    - Some big companies, like Amazon and Palantir, do not negotiate new grad offers. But if your recruiter does not mention not being able to negotiate, it is fair game.

Random personal experience interviews and timelines (for reference)

These are interview timelines from 2019 to 2022 (mostly 2020 and 2021), so most of these should be relevant for upcoming interview cycles.

Intern - Specific programs

- Google (STEP intern)
    - Sept 19: Applied with referral
    - Sept 27: 2 back-to-back (technical, each 45 minutes)
    - Oct 28: Offer (2 weeks to decide)

- Microsoft (Explore intern, pre-pandemic)
    - September 24: got a first-round interview (schedule to October 11), leetcode easy
    - October 16: got the in-person onsite interview, but there weren't any spots before winter break.
    - January 24: free trip to Seattle for a final onsite interview with 2, 1h rounds. One had a PM focus, the other behavioral, and an easy coding question.
    - January 27: offer
    - Return offers: usually week 11/12 of internship, 2 (now 6)-week deadline to accept.

- Palantir (Path Intern)
    - Oct 11: Applied (I think this was the deadline actually)
    - Oct 13: Received hackerrank, completed Oct 21. (Reasonably easy, string edit distance)
    - Oct 28: Heard back from the recruiter.
    - Nov 10: 3 back-to-back interviews (coding, learning, design)
    - Nov 17: Call with the hiring manager.
    - Nov 26: Received offer. (Don't remember how much time I had to decide)

- Palantir (Path Intern)
    - August 23: Applied, received an OA (hackerrank, 50 minutes) immediately
    - September 17: Scheduled for a recruiter call (behavioral with a heavy emphasis on why Palantir)
    - September 21: 3 back-to-back technical round (tech, decomposition, and learning)
    - October 5: Hiring Manager round (tech with more emphasis on behavioral)
    - October 19: Second Hiring manager round (not normal to have 2 hiring manager rounds) (decomp with more emphasis on behavioral).
    - November 2: Offer (2 weeks to decide)

Intern - General SWE

- Salesforce (SWE intern)
    - June 14: an invitation to apply for early access, application
    - June 15: coding challenge (Hackerrank with 2 medium problems, 60 min)

    - July 14: SWE expedited “onsite” with 2 interviews, one was with the hiring manager and was purely behavioral, and the other had a systems design focus.
    - July 26: interviews in the morning and offer later the same day (~~2~~ negotiated 3 weeks to respond as I was waiting to get my return offer and hear back from Uber)

- Uber (SWE intern)
    - July 12: got the first round technical interview (Codability, 60 min)
    - July 26: first interview (leetcode medium with a hard follow-up)
    - July 27: moved forward to the final interview
    - August 2: final interview (leetcode hard)
    - August 16: offer call, I think I was given 3 weeks to decide

- Microsoft (SWE intern)
    - July 17: Applied with a referral
    - October 7: Recruiter call: basic questions about computers and behavioral
    - November 9: 2 back-to-back technical round (both leetcode medium) (online)
    - November 17: Offer (2 weeks to decide)

- JPMorgan (SWE intern)
    - October: applied, no referral. Received a 2-hour HireVue interview (with OA) and a personality assessment.
    - November 9: 2 back-to-back technical + behavioral interviews (lots of general CS questions, i.e. how much do you know about blockchain, etc…）
    - Nov 15: Offer (2 weeks time to decide)

- C3AI (SWE intern)
    - October 5: Applied with referral
    - October 26: Received 2-hour OA
    - Nov 15: First round: 45 minutes (technical)
    - Nov 19: Second round: 2 back-to-back (technical, each 45 minutes)
    - Nov 30: Third round (technical, 45 minutes)
    - Dec 13: Fourth round: 2 back-to-back (slightly technical + behavioral with manager and VP)
    - Dec 17: Offer (3 days to decide *exploding offer)

- Amazon (SWE Intern)
    - Sept 2: Applied.
    - Sept 25: Received assessment. Completed Sept 29. (debugging)
    - Sept 30: Second assessment. (Java object-oriented design) Completed Oct 4.
    - Oct 5: Third assessment. Work simulation.
    - Oct 28: Interview availability. (Expired Nov 8)
    - Nov 4: Interview (simple coding question)
    - Nov 10: Offer. (Expired Dec 1).

New grad

- Google (New Grad, SWE)
    - August 30: Applied with referral
    - November 9: Received OA
    - January 25: 1 Googliness Interview, 4-back-to-back technical interview round (Leetcode mediums and leetcode hards)
    - February 7: Passed Technical Interviews, Fill out GMatch
    - April 8: 2 Team Match Interview/Hiring Manager Interview
    - April 20: Offer (10 days to decide)

- Amazon (New Grad, SWE)
    - August 18: Applied with referral
    - November 8: Received OA
    - December 16: 3-back-to-back technical round (Leetcode medium)
    - December 21: Offer (3 weeks to decide)

- Amazon (New Grad, SWE)
    - July 1st: Applied online, no referral
    - August 1st: OA 1 completed on August 7, got and did OA 2 same day
    - August 8th: Interview invite for 30 minutes (in 2022-2023 you get either 1x30 min interview if you aced the OAs or 2x60 min if you didn't).
    - August 22: Interview.
    - August 25: Offer (2 weeks to decide)

- Blue Origin (New Grad, SWE)
    - August 20: Applied
    - October 6: Technical Phone Screen
    - November 4: 3 back-to-back technical interview (Leetcode medium and leetcode easy), 1 hiring manager interview, 1 PowerPoint presentation
    - November 11: Offer (10 days to decide)

- Meta (Facebook) (New Grad, SWE)
    - Applied with a referral in August
    - Oct 5: First round: 45 minutes (technical)
    - Oct 25: Final round: 45 minutes, behavioral
    - Oct 28: Final round: 2 back-to-back (technical, each 45 minutes)
    - Nov 9: Offer (2 weeks to decide)

- Brex (New Grad, SWE)
    - Reached out by recruiter in July
    - Jul 16: Received Codesignal OA
    - Jul 27: First round: Codesignal, technical
    - August 11: Final round: Coding, Debugging, Behavioral

    - August 11 (Same day): Offer (2 weeks to decide)

- Oracle OCI (New grad, SWE)
    - July 18: Recruiter reached out on Linkedin and scheduled a screening. Simple behavioral wanted to know about past internships/projects/classes and asked what I knew about Oracle.
    - Sept 8: Invited to final round interviews (skipped the tech screen)
    - Sept 14: 3 back-to-back 60 min interviews. First was an initially easy leetcode, with follow ups from medium and hard questions with a strong emphasis on the testing component (spend most of the time writing unit tests, assertions, and throwing exceptions), second was a medium/hard system design involving airports and scheduling (design API, database and storage/ds, complexities) and third was 2 medium questions about N-ary trees/dfs/bfs. All had at least 10 min of behavioral questions.
    - Sept 26: Offer, 2 weeks to accept.

Contacts

Feel free to contact us with questions or for more information!

- PJ Sangvong ‘23 (Palantir) [Linkedin]
- Luisa Escosteguy ‘23 (Microsoft) [Linkedin]
- Louis Ye ‘22 (Google) [Linkedin]
- Shiyue Zhang ‘22 (Brex) [Linkedin]
- Juanito Zhang Yang ‘22 (Palantir) [Linkedin]