Jake Gavin, James Harkins, Zach Imholte, Ryan Moran, Ryan Tick
CSE 40175
Project 2
Due: 2/18/16

Job Interview Guide

When to Start Preparing

The majority of students wait too long to start applying for jobs. Some wait until the second semester of their Senior year to even beginning thinking about their life after college. Why would you procrastinate when it comes to choosing a career?

The best time to start preparing is arguably before your Junior year even begins. It is only fair to start at the internship phase of college. Most students, at least at Notre Dame, will get an internship related to their major the summer of their Junior year. If you are lucky, this internship can lead to a full-time opportunity. Because of this, it is extremely important to really prepare for the internship process.

With that out of the way, let's assume that you didn't get an offer from your summer internship or you turned it down. While it is unfortunate, it's definitely not the end of the world. Starting during the summer going into your Senior year, or beginning during the start of the Fall semester will guarantee you have enough time to search thoroughly. This is probably one of the biggest decisions you will have had to make so far, so it's important to give yourself enough time.

How to Prepare

Preparing for the Behavioral Interview

The absolute most important thing to get right in a behavioral interview is to establish a rapport with your interviewer. Smile, make eye-contact, show interest in your interviewer's life and their job. Most interviews have been decided within 5 minutes of the beginning of the interview (I posit that the "tell me about yourself" and the "Why do you want to work for ABC Company" questions usually fall in this time slot, in addition to the chit-chat where you attempt to establish a rapport), so please use those precious minutes to establish a connection with your interviewer. But for the remaining 55 minutes, there's still a lot of ground to cover to affirm your interviewer's already positive impression.

The behavioral portion of the interview is often underemphasized when people discuss computer science interviews. However, if you do not ace the behavioral interview questions, it is unlikely that you will receive a job offer. Think about it: would you rather work with someone who is technically brilliant but is painful to work with, or someone just as technically competent as you but who is incredibly easy to work with? Almost invariably, a group of smart people working together will output a quality of work directly proportional to how well they work together. So that's your job: to convince your interviewer that you are a dream to work with.

You do this by researching the position and company, reflecting on past experiences, and practicing.

First, researching the position and company is important because you want to be able to speak intelligently about why you are a good fit for the role. If the role calls for someone who is good at working in large groups under frequent deadlines, you will have to highlight that in your interview responses. Additionally, you will have to answer the question "Why do you want to for for ABC Company," even if you aren't explicitly asked the question. This requires having done a lot of research about the company so that you can adequately answer the question. Below I have detailed an example answer to the question "Why do you want to work for Google?" and it will serve as a good framework for how to answer any "Why" type behavioral questions.

Framework for Answering "Why" Questions

1. Answer in Threes
2. Summarize at the beginning
3. Count out loud

An example answer one could give for Google: "Well, there are three reasons why I want to work for Google: their mission, the opportunity to learn, and product management. First, I want to drive Google's mission to provide universal access to information, because equal access to education is something I have developed a passion for. Second, I am looking for an opportunity to invest heavily in myself by tackling a steep learning curve, and Google will provide the opportunity and resources to challenge myself outside of my comfort zone. Third, I want to own a project, and see the impact that my work has on users, so Google's Associate Product Manager program is perfect for me to leverage my love of products while allowing me to learn how to focus on the user."

You can tell that the answer gives three reasons (1) and that I summarized those three reasons before going into detail (2). This is because when you are speaking in an interview, this makes it much easier for the person to understand what you're saying - which is also why I suggest you count out loud (3), so that the person knows which of the three reasons you are detailing at that moment. It feels awkward at first, but it makes your answers much easier to comprehend and internalize for the interviewer. Remember, you want the interviewer's job to be as pleasant as possible. But most importantly, the best answer to "why do you want to work at ABC company" is an answer that only you can give. Figure out reasons that are unique (or somewhat unique) to you, and give those answers.

So that covers why you need to do your research: so that you can convince the company that you are a good fit. Similarly, you need to reflect on your past experiences to identify STAR stories that you can tell about yourself, as well as prepare an answer to the "tell me about yourself" question. Your answer to this question should invariably be about 30 seconds long. This is your opportunity to present yourself as an individual who will be remembered by the interviewer after the interview ends, so please be sure to do yourself justice: be succinct (30 second response), be impressive (highlight your most impressive feats), and be memorable (be unique).

The STAR stories are stories that highlight your most impressive and relevant accomplishments, and they are called STAR stories because the story should be told in order of Situation, Task, Action, and Result (similar to the CAR - context, action, result - method). You will be asked a lot of questions starting with “Tell me about a time when…”, and the … are often, but not limited to: you had to overcome an obstacle to get something done; you had to convince someone to go along with your plan when they disagreed with you; you had to work with a deadline that was moved up on you; and countless others. This is why it is so important to reflect on your past experiences, so that you can have a bank of 5-10 experiences to draw from that reasonably could answer any question of this type. The framework for answering is STAR:

1. Situation: You first want to give the interviewer the situation (start with saying “The situation was…”)
2. Task: You want to frame the experience in a way such so that the interviewer is aware of the explicit task you were given. This provides them with more context as to what specifically your goals were for the project (the task can be given by a professor or manager, or self-defined, or both). Again, start with “The task I was given / the goal I set for myself was…”
3. Action: What specifically did ***you*** do for your team to accomplish your goals. Don’t talk about what the group did; what specific actions did you undertake that allowed the group (or if not a group project, yourself) to be successful in this endeavor?
4. Result: Here is where you get to brag about how incredible the results of those actions were, and compare them with the task you were given. Again, start with saying “the results were…”

You will notice that I suggest you speak aloud to delineate which part of the STAR method you are detailing at that time. This is for the same reason as counting aloud when listing reasons for a “why” question: it makes the story easier to follow for the interviewer. So reflect on your past experiences, figure out what experiences you have had that coincide with what the company professes it desires (in the role description and on the “about us” portion of the website), and then come up with 5-10 STAR stories that highlight the best performance you’ve ever achieved.

Lastly, it helps to practice. You can do this by going through interviews with companies that you are less concerned about acing the interviews for, or by setting up mock interviews with the career center. You just don’t want the first time you “tell me about yourself” or tell a STAR story to be the most important time.

In summary, be sure to establish a rapport with your interviewer. Then be prepared to ace the rest of the interview by knowing the company, knowing the role, and knowing yourself.

Preparing for the Technical Interview

When preparing for a technical interview where you have to code or talk your way through a wide variety of technical questions and interview formats, it can seem like there's no way you can possibly be prepared for everything the different companies you will be interviewing with are going to ask you. However, you shouldn't have to feel like you need to familiarize yourself with every technical interview question ever asked by Google, Facebook, and Microsoft. Instead, focus on fundamental interviewing skills that can be extrapolated to conquer anything thrown your way. There are three keys to being prepared for technical interviews which can be so diverse:

1. Have a very solid grasp on at least one well accepted programming language
    - This is important because in most technical interview environments you won't be able to use resources such a textbook or stackoverflow to look up things such as syntax or function usage. Showing proficiency in this way suggests to the interviewer that you are an experienced programmer.
2. Understand the computer science fundamentals
    - Data structures, operating systems, Big-O notation, etc. All of the boring fundamentals you've learned over the years will likely come up in a big way during the technical interview process. If you can code a hash table with your eyes closed in practice, it's going to be a lot easier to do it on a whiteboard at Google HQ.
3. Understand and practice general interviewing tactics
    - You can know C++, data structures, and operating systems like the back of your hand, but you're not going to impress an interviewer if you don't know how to give an answer in the interview environment. For example, interviewers love when you ask questions, explain your code while you're writing it, and in general write very clean, readable code. This largely comes down to practicing your communication skills, but it is critical in being able to convey your skill set to an interviewer.

The most-general way to practice the skills listed above is to practice writing code in a timed setting where you have no resources, such as writing code into a word document or on a whiteboard. Be sure to write code that you have solutions for so that you can check your solutions for correctness. Sites such as www.hackerrank.com are a great resource as you can write and compile code in a web browser. This site in particular has a relatively diverse set of problems to work on that reflect what you might encounter in a technical interview. Another similar site is www.interviewcake.com. Just be sure that you know the fundamentals (there are many resources for fundamental listed below), be able to effectively write code without using Google, and have practice explaining your code as you write it. When you feel like you've got a good grasp on the three keys listed above, you've got a good chance to nail any technical interview that you walk into, no matter the topics of the questions they ask or the format in which you have to present your answers.

What Resources are Available

Effectively using the resources available to you is vital to succeeding in the job interview process. There are a variety of different types of resources one should strive to harness throughout various steps of the process.

First and foremost among important resources that should be utilized are one's own peers, relationships, and connections. When undertaking the interview process, it is immeasurably valuable to seek the advice of both those who have come before you and those currently going through the same process. Talk to alumni in your field. Talk to friends interviewing for the same types of positions and companies. Talk to anyone and everyone who may be able to give you valuable tips and pointers - both general and specific. Many universities such as Notre Dame even have professional and/or student lead career centers and services whose job is specifically to offer a variety of advice for those with upcoming interviews. The professionals at a career center can assist with preparation for every step of the interview process - from building a resume to practice for specific behavioral and technical questions. A mock interview at the Notre Dame Career Center is probably the closest experience to a real interview one can get.

Another category of resources are those that help you prepare for specific questions and types of questions that may be asked of you. Dozens of books and websites exist solely for the purpose of aiding in interview preparation. One of the staples of any computer science student preparing for a technical interview should be *Cracking the Coding Interview: 150 Programming Questions and Solutions* by Gayle Laakmann McDowell. This book focuses on specific types of technical interview questions asked by companies like Google and and Apple in interviews for top software positions. Many students confident in the knowledge gained from their classes are nonetheless stumped when faced with an unfamiliar technical question in a time-sensitive interview environment. Odds are these preparation resources will not contain the exact questions you will encounter in a live interview, but simply seeing similar types of questions in advance can significantly increase one's confidence and ultimately one's performance in a high-pressure interview environment.

Lastly, one should take care not to neglect the resources specific to the company and position being interviewed for. One should always take the time to learn as much about a company and position as possible before an interview. Company websites and job postings often contain essential information that may come up in an interview. Something as simple as being knowledgeable about a company's history and achievements can tip the interview scales in your favor.

Which Extracurriculars to Undertake

A lot of students have to work during school to make money. If you're one of those students, then it may be hard to fit in extracurriculars (ECs) to your busy schedule. However, if you can find the time, extracurriculars can both benefit you and look very appealing to interviewers.

There are three different types of ECs. The first are those that complement your academics. For example, you may do research or working in a computer cluster in order to build upon the skills you've learned in class. The second type are those that help you build skills that aren't academic. Maybe you joined a debate team so learn to speak/argue more clearly, or perhaps wrote for your school paper. These ECs are no less valuable than the first type, and they help you build valuable skills that companies will find appealing. The last form of ECs are those that you do for fun; maybe they don't necessarily look as attractive on paper as the first two types, but they are enjoyable to you and make you unique. If you're lucky, you might even be able to combine all three types of ECs into one. For example, working as a teaching assistant for a class you previously took can reinforce your academic abilities, help you build other skills such as public speaking, and even provide a bit of fun if you enjoy helping your peers. Getting paid as a TA is just a bonus on top of these other benefits.

It's important to have a good mix of all three if you can. Each will offer their own benefits/talking points in an interview. But more importantly, you want to show that you're a leader and you are driven. You *should* ideally be pursuing these ECs for your own benefit, because you enjoy what you're involved in and not because they look good on a resume. With that being said, even if you have broad interests, try to focus on a few ECs. Don't spread yourself too thin, where you are barely involved in any club/hobby.

How to Network

Networking is a very difficult task for upcoming college graduates. To this point your career has been relatively short and you probably do not have a lot of professional contacts. You may have some through previous internships or part-time jobs in the past, but your network is certainly smaller than it will be in the future. This leads many students to shy away from networking as they feel that it will be pointless; they have no idea where to begin.

For most students the best place to begin networking is through their university. This may be difficult depending on how strong your school's alumni network is, but for Notre Dame Computer Science and Engineering students this is not an issue. You have an extremely strong alumni network to connect to, and the school does a great job connecting students to alumni. You really don't need to put much effort into finding alumni connections yourself. You just need to pay attention to the networking opportunities that the university runs during the year. You get a lot of emails on a day-to-day basis, but don't ignore those emails about networking opportunities!

Look for the opportunities that match with your interest best, there are sure to be many of them. The alumni there will be glad to network with you. These are a great way to build your alumni network with very little effort.

How to Understand Contracts and Negotiate Compensation

This is one of the most difficult parts of the job finding process, as many students have never had to go through this stage before. This part of the process is significantly more crucial when looking at your first full time job, rather than your previous internships and part-time positions. It's easy to get too focused on your salary and forget that it is only a part of your compensation. It's also easy to miss key terms in your contract that will greatly affect your life in the future, such as non-compete or non-disclosure agreements. There's a lot more to take into account then your starting salary.

In terms of understanding your contract, the best thing to do would have it looked at by someone who is a professional. Not that you necessarily need to hire someone, but try to find a friend or relative that is used to dealing with contracts such as a lawyer. If this isn't possible, there are many resources online that can help you with this process. Usually the language isn't too confusing, but you need to make sure that you read and understand every part of the contract. Look out for clauses that could affect you even after employment, such as non-compete clauses. Not that you need to completely avoid these, as they are necessary in some fields, but take them into account. You may even want to look up your state's laws on non-competes to see to what extent they can be enforced. Also read through your contract to come up with a total compensation number, which includes a lot more than your salary. Look for medical or dental benefits, 401k matching, etc. Try to come up with an actual number that can be used when comparing jobs.

Negotiating compensation is a very difficult task, especially for your first-job out of college. It's easy to feel overwhelmed and that it is not your place to negotiate your first job out of college. This part of the process should be done on a case-by-case basis. Some companies will be open to salary negotiation, other's will have a set salary for a position that cannot be negotiated. Even with this second group of companies, negotiation is still an option. If you feel that the salary isn't high enough, you can often negotiate other forms of compensation. You may be able to get more days of paid leave or other benefits. Negotiation usually doesn't hurt to try if you remain respectful through the process.

Things I Wish I Knew as a Freshman

As a Freshman - and even as a Sophomore and Junior - I wish that I knew how difficult technical interviews would be regardless of how well I did in my classes at Notre Dame. Had I known this in advance, I would have started the process of practicing and preparing for special types of technical interview questions much earlier. *Cracking the Coding Interview*, as mentioned above under the resources section, is a dense book. And to get the most out of it, one really needs to take the necessary time to attempt and understand each question it poses. I wish I had known that preparing for technical interviews is a job unto itself.

- Ryan Moran

There's a difference between getting good grades in class and truly understanding the material. A lot of my interview questions probed on things that I learned through my extracurriculars, and I only got the answer right/felt confident during the interview because I did work outside of class. Additionally, I think it would have been helpful to have talked to a senior about all this stuff, as a freshmen -- maybe something like a Big Brother/Big Sister type of departmental thing. There's so much that I've learned during the whole process that I know Freshman me would have loved to know.

- Ryan Tick

Having good grades and being able to perform well on technical questions in the heat of the interview environment is certainly important, but forming good relationships with alumni and interviewers is as important or even more so. The Career Fairs are great for establishing relationships, but don't wait until then to begin talking to companies you may be interested in. You'll receive hundreds of emails about events such hack-a-thons and guest speakers from great companies that are great opportunities to form good relations with people who can help you get your dream job. I wish I hadn't taken a lot of these opportunities for granted.

- Jake Gavin

There are certain classes that will be heavily stressed during your technical interviews. I wish I knew how important these classes were while taking them, as opposed to months or years after. It's really easy to get good grades in these classes without actually understanding or remembering the material. When it comes time for technical interviews, you will be glad that you took the time to understand rather than memorize. It will make preparing for and succeeding in these interviews so much easier.

- James Harkins

The biggest thing I wish I knew before starting this process is how important it is to practice, and practice well. My interviewing performance got so much better after completing dozens of practice and real interviews, and I wish I would have known how much better I could have been previously had I only practiced. When I say it's important to practice well, I'm referring to practicing in the right format, and practicing over time. Doing two practice interviews every week for a year mounts up to over 100 interviews, which is hard to beat. I really wish I would have tapped into the power of doing a little of something every day much earlier in my career. Now it's a tool that I'll take with me for the rest of my life.

- Zach Imholte