What is the Technical Interview?
What You Need to Know about the Gatekeeper to CS Careers

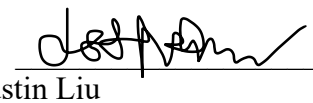
A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science
University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science, School of Engineering

Justin Liu
Spring Semester 2022

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Signature  Date 05/07/2022
Justin Liu

Approved _____ Date _____
David G. Graham, Department of Computer Science

What is the Technical Interview?
What You Need to Know about the Gatekeeper to CS Careers

CS4991 Capstone Report, 2022

Justin Liu
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
jl8wf@viginia.edu

Abstract
The technical interview, TI is the gatekeeper to many computer science careers, and yet it is not fully understood nor taught in higher education. Having done research on the current state of higher education, I have concluded that there must be changes in how students are prepared for industry.

First, in order to prepare for the TI, one must know what it is. Second, the interviewee must understand what is expected in a TI. To determine this, I interviewed recent computer science graduates to get their companies' perspectives about what they want from their potential employee.

Students will have easier time in getting a job that they expect from attending a university or college. Future recommendations can range from experimenting with new teaching styles to researching the actual purpose and benefits of the TI itself.

1. Introduction
"College is like a fountain of knowledge and the students are there to drink", according to Palahniuk, an American journalist [7]. If we were to take this at face value, what kind of knowledge should the fountain provide to students? For computer science students, the knowledge has to be applicable in the field and useful in getting the job.

Demand for computer science related jobs is projected to explode at a tremendous rate [6]. Many students will major in computer science in the hope of getting a job in this growing field; however, a TI is a part of the process of getting hired, almost becoming synonymous to a rite of passage into the computer science field.

The term "TI" is misleading because it is not like the typical interview where you are asked questions. A TI is a "test of the candidates' problem-solving or 'analytical ability'" using a coding problem/challenge [1]. Think of the TI as a combination between an audition, similar to how musicians might apply to join an orchestra, and a high stakes test, where not succeeding or passing often means not securing a job. The applicant is also expected to speak aloud about their thinking process in solving the given coding problem. This means the TI not only requires the applicant to be proficiently skilled, but also articulate and able to communicate clearly about what he or she is doing to solve the problem.

Unfortunately, higher education does not adequately prepare its computer science students for the TI. Though universities teach

their students how to code and the concept of efficient algorithms, but many students are often left on their own to learn what exactly the TI is. Students who attend higher education with the expectation that it will help them land a job will be sorely disappointed when they find out that for the past four years, they have been learning concepts and theories that are not applicable in getting a job, unless that job is in research and academia.

2. Related Works

The TI is a relatively new tradition in the hiring process of software engineers and other computer science- related fields, so there are a limited number of papers on the topic [4]. However, [3] clearly states that "software engineering job candidates are not succeeding at TIs" because applicants are mistaken in what is expected of them during the TI. [3] acknowledges that candidates are proficiently skilled, but do need to work on interpersonal skills that would allow them to properly communicate how they are solving the problem.

[2] found that "only 65% to 70% of IT students manage to obtain jobs after graduation." They attribute this in part to "recession, lack of qualifications and unrealistic expectations." However, fresh graduates are also under-performing in interviews contributing to the deficit. It should be noted that when [2] talks about interviews, [2] does not specifically mean TIs but job interviews in general. Though not explicitly about the TI, [2] shows that there is unpreparedness in college students when it comes to interview-like situation.

3. Proposed Project Design

I propose modifying the CS curriculum or adding a course at UVA that addresses the lack of exposure to the TI. In this way, UVA students will be given a level playing field for facing the TI. Taking the input of recently graduated CS majors from UVA, I have come up with guidelines for such an implementation.

3.1 Responses from Recent UVA CS Graduates

I interviewed three recent CS graduates who have been working for a little less than one year in well-known companies to get their input. The results of the brief interviews are shown in Figure 1 below.

3.1.1 How to Prepare for the TI

Although all of them are different, all of them heard about the TI only by word of mouth or when they were on the job hunt themselves. There was general consensus that the addition of a course on the TI would have been welcomed while they attended UVA.

College	SWE 1	SWE 2	SWE 2
Works for…	***Amazon***	**Capital One**	**Appian**
How did you find out about the TI?	Word of Mouth	Word of Mouth	Word of Mouth
How did you prepare for TIs?	**Leetcode, Cracking the coding interview**	**Leetcode, reviewing former questions**	**Leetcode, taking breaks**
What does your company use the TI for?	Assurance of fundamental knowledge weed out process	Identifying collaboration, good grasp of fundamentals	Identifying collaboration, communication, thought process
Would you have liked a course that taught the TI when you were in undergrad?	**Not a full-blown course, but a seminar would have been nice**	**Yes, for sure, it would have been convenient, but only 1-2 credits**	**Yes and no, 65/35 split, good for those who aren't that connected**

Figure 1. Interview questioned asked

From the responses from Figure 1, preparing for the TI boiled down to preparing ahead of time. Leetcode and Cracking the Coding Interview were the prime resources for preparing for the TI. Leetcode is a website solely dedicated to preparing students for TIs, testing its users with real questions from companies such as Google and Microsoft. Cracking the Coding interview is a book that teaches algorithms most commonly seen in Tis, which would help familiarize students with the process.

In addition to these, practicing with a peer helps develop the communication aspect of the TI by allowing one to explain their thought process to someone instead of talking to oneself. Communicating in a TI is often looked over by those recently exposed to the TI and is highly stressed in TI workshops.

3.1.2 The Do's and Don'ts of the TI

Being able to solve the problem is a given in the TI. However, there are many different ways to solve a given problem. Some are clearly better than others in saving the amount of space in memory used and how fast the algorithm is implanted. First-time interviewees often focus too much on trying to solve the problem using the most optimal solution. Sometimes they can, but if they don't, they end up wasting a lot of valuable time. Showing you can solve the problem is already a plus for the interview.

The most confusing part of the TI is in the name. It is not actually an interview. Solving a given problem is straightforward. Having to explain the thought process involved in solving the problem under a time constraint in a high stakes process is not. It is helpful to talk to the interviewer about what you are doing. Do not try to do it yourself in an attempt to show you are independent.

3.2 Ideal Solution

The CS 9: Problem-Solving for the CS TI course at Stanford University is exactly the kind of course I had in mind for UVA students [5]. Having a course that addresses post-graduation goals is necessary for any student. Preparing students to actually enter the field is just as important as the knowledge needed to stay in the field. The course CS 9 is a one credit course, which means the workload should be relatively light. CS 9 is taught by those with many years of experiences in the industry. This is an advantage compared to most courses as most professors have been disconnected from industry.

4. Anticipated Results

The solution I propose for UVA's lack of exposure of its students to the TI is an elective one-credit course that teaches about the TI. The course would be very similar to CS 9 taught at Stanford. The course would give students the basic knowledge they need to succeed by going over algorithms that are common in TI, and helping them get accustomed to speaking their thoughts aloud as they solve a problem, and preparing them overall. In essence, the course would combine Leetcode and Cracking the Coding Interview to prepare students in a higher education setting.

The predicted results of this course would be that students who take it will be more confident and better equipped to succeed in their TI. UVA's CS major employment rate after graduation should increase, corresponding to those who have taken the course. This in turn would promote UVA's reputation as a school that prepares its students for the workforce.

The success of the class will be measured by asking students to score on a scale of 1-5—with 5 being significant improvement and 1

being little to no help—the degree to which the course enabled them to succeed. The student will also be asked if they are actively seeking employment or not.

5. Conclusion

The TI is the gatekeeper to the CS career field. Students will face the TI eventually if they intend to find employment in the CS field. Unfortunately, students usually have to find out about the technical interview through word of mouth or the hard way when they start the job hunt themselves. Knowing that the TI is not so much an interview but an audition where you show your thought process goes a long way for those who were not even aware the TI existed. If higher education offered an optional 1-credit course that taught about the TI, students would be better prepared for this challenge.

6. Future Works

The next step can be what exactly should be included in such a course. Future works can also include the pros and cons of the technical interview and effects of the technical interview on employment and quality of employees since it was introduced. The technical interview itself might be broken according to [1], so a potential future work might be what could replace it or what can be improved.

8. Acknowledgments

I would like to give acknowledgement and thanks to Professor Vrugtman for her help in advising this paper. I would also like to give thanks to my friends, Ryan Kann, Daniel Tran, and the third respondent, who wishes to remain unnamed, who recently graduated and were willing to be interviewed.

References

[1]Mahnaz Behroozi, Chris Parnin, and Titus Barik. 2019. Hiring is Broken: What Do Developers Say About Technical Interviews? In 2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 1–9. DOI:https://doi.org/10.1109/VLHCC.2019.8818836

[2]R. H. A. Dulmini H. Ehaliyagoda and Laurianne Sitbon. 2018. A proposed system to increase work readiness of fresh IT graduates at interviews. In Proceedings of the 30th Australian Conference on Computer-Human Interaction, ACM, Melbourne Australia, 443–447. DOI:https://doi.org/10.1145/3292147.3292182

[3]Denae Ford, Titus Barik, Leslie Rand-Pickett, and Chris Parnin. 2017. The Tech-Talk Balance: What Technical Interviewers Expect from Technical Candidates. In 2017 IEEE/ACM 10th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), 43–48. DOI:https://doi.org/10.1109/CHASE.2017.8

[4]Marvin Wyrich, Daniel Graziotin, and Stefan Wagner. 2019. A theory on individual characteristics of successful coding challenge solvers. PeerJ Comput. Sci. 5, (February 2019), e173. DOI:https://doi.org/10.7717/peerj-cs.173

[5]CS 9: Problem-Solving for the CS Technical Interview. Retrieved February 16, 2022 from https://web.stanford.edu/class/cs9

[6]Computer and Information Technology Occupations : Occupational Outlook Handbook: : U.S. Bureau of Labor Statistics. Retrieved October 9, 2021 from https://www.bls.gov/ooh/computer-and-information-technology/home.htm

[7]2021. 40 College Life Quotes to Get You Through the School Year. *Seventeen*. Retrieved March 29, 2022 from https://www.seventeen.com/life/school/a28799474/college-life-quotes/