Potential Questions

1. Why did I choose journals for my English publications instead of top conferences, as is more common in the CS community?

Submitting to top conferences posed significant challenges. Iranian universities typically restrict such submissions to Ph.D. students, with limited financial support. To provide perspective, the entire funding for my research was less than the cost of a single TOEFL exam.

Opting for specific journals was a choice in consultation with my supervisor, Professor Jalili. To ensure that my papers would be published when deemed valuable by experts in the field, in addition to examining journals' scope, quartile, and impact factors, I also examined their editorial boards to confirm their expertise in the areas relevant to my work. Moreover, I purposefully suggested potential reviewers, providing detailed reasons to support my recommendations within the journal's guidelines.

2. What is your self-assessment of the quality of papers or general research projects you participated in?

My self-assessment of the quality of research papers and projects I engaged in varies due to my early exposure to research and the diverse responsibilities I managed independently. Some projects delivered strong results, reflecting my growing research skills. However, given the multifaceted nature of my tasks, not all reached their full potential. These experiences, though diverse, collectively contributed to my development as a researcher.

3. Regarding GitHub repositories linked to my papers: Why are they relatively small in size, containing only a few folders and files?

While it may seem that my repositories are minimalistic, it's important to note that I have undertaken various types of work over the years. Some of my older public projects, like 'Harif,' which I developed as an undergraduate, consist of over 6,000 lines of code. These projects involved implementing numerous standard algorithms and data structures from scratch.

During my research, I often needed to replicate other researchers' work quickly to assess its applicability to my own projects. This sometimes led to encountering messy repositories that took significant time to decipher. However, I realized that the replicated work didn't fit my project's requirements.

Given my solo work and the impracticality of cleaning up all my code, I faced two options. I could either store all my code, experiments, temporary files, etc., in a repository in a disorganized manner or choose to include only the minimum but clean code necessary to replicate the paper's main results. I chose the latter approach, with the belief that this strategy aligns with Dijkstra's wisdom: 'I mean, if ten years from now, when you are doing something quick and dirty, you suddenly visualize that I am looking over your shoulders and say to yourself 'Dijkstra would not have liked this,' well, that would be enough immortality for me.

4. Some parts of your code seem to be written simplistically. Aren't you aware of more advanced features offered by the language?

Yes, I am familiar with more advanced features provided by programming languages. However, at times, I provide assistance to friends in both academic and industrial settings through online consultation sessions. This assistance often involves tasks such as tailoring an existing codebase to their specific requirements, aiding with code generated using AI tools, or conducting comprehensive reviews and offering feedback on their code and reports. Drawing from my experience, I have found that simpler code tends to be more comprehensible and user-friendly for a broader audience.

5. But, why did we find that not all of your GitHub code repositories adhere to clean code practices during our review?

The evolution of my coding practices over time plays a significant role in this. During my undergraduate years, some of my code tended to be less structured and more intricate. This was primarily due to the demands of working on complex projects with multiple responsibilities, which included tasks ranging from algorithm design and software architecture to actual implementation, all within tight deadlines.

Experience has been a crucial teacher in refining my coding style. As I gained more exposure to software development and project management, I gradually adopted cleaner and more efficient coding practices. It's important to recognize that coding is a journey of continuous improvement, and my commitment to writing clean code has grown with each project and learning opportunity. Please take note of the timestamps for each project, not only on GitHub but also on my website, where I have provided more precise timing!

6. Can you elaborate on your coding experience beyond the publicly available projects on your GitHub? Please share details about any additional algorithms or data structures you have implemented from scratch?

I've been involved in a variety of projects, both self-funded and commissioned by companies and individuals. I've provided an overview of some of these projects on my website, where I often played multiple roles, including software design and implementation.

Regarding the latter part of your question, I have extensive experience in implementing a wide range of standard algorithms and data structures using various programming languages. In some cases, I opted to develop them from scratch, even when existing implementations were available, as it served as valuable practice and deepened my understanding of these concepts. Additionally, there were instances where I couldn't find suitable existing implementations or encountered challenges with them, such as steep learning curves, inadequate documentation, limited customization options, or security concerns.

Below, I've compiled a list of some of the algorithms and data structures I've personally implemented, and you can also find some of these implementations in my GitHub repositories:

- Machine Learning: Linear & Logistic Regression, Decision Tree, KNN, Random Forest, AdaBoost, Naïve Bayes, and KMeans in Python
- Artificial Intelligence: Monte Carlo Tree Search in Python
- Nature-Inspired Optimization Algorithms: Genetic, Tribe Particle Swarm Optimization, and Discrete Grey Wolf in C# and Python
- Graph Algorithms: DFS, BFS, Prim, and Kruskal in C#
- Sorting Algorithms: Bubble Sort, Merge Sort, Quick Sort, Insertion Sort, Heap Sort, and Counting Sort in C#
- Games: Tic-Tac-Toe, Chess, Raichu, Poker Squares, Puzzle, Snake Game, and Typing Game in C#, Java, C, and Python
- Compiler: Recursive Descent, LL1, and others (I may not remember all of them) in C#
- Signal Processing Algorithms: Pan–Tompkins algorithm in MATLAB
- Data Structures: Singly, doubly, circular array-based, and pointer-based linked lists, stack, and queue; binary, binomial, and Fibonacci heaps; disjoint-set forests; binary search trees; adjacent matrices and adjacent linked list graphs in C#.

Please note that while I mentioned you can find some of these implementations, not all of them are available for public viewing. If you are interested in checking out a specific implementation of an algorithm or data structure, you can request it, and I will do my best to locate and share it with you. However, it's essential to keep in mind that since I've implemented these algorithms over time, there is a possibility that I may not be able to locate some or many of them, except for those shared on my GitHub or written in recent times.

In such cases, if you have genuine concerns about my implementation skills and it's a critical factor in your decision-making process among candidates (i.e., you've considered all other factors, and this factor could be a game-changer), you can select one or a few algorithms or data structures from the list I provided. Please allow

me some time to check my references and refresh my memory, and I will implement and share them with you. However, I kindly request that you consider this option only if you have a significant concern, as it would require a considerable time investment on my part. I believe in fairness and efficiency in the evaluation process, and I'm happy to provide additional information when it truly matters.

7. Can you tell us what available tools you have worked with or have customized for your use cases? Specifically, can you elaborate on your computer skills in your CV?

I have experience working with a wide range of tools, some for specific purposes and others more extensively. While I cannot list all of them on my CV due to space constraints, I want to highlight my proficiency in certain languages and tools.

When I describe myself as 'highly skilled,' it means that I have in-depth knowledge and hands-on experience, even if I may not use that language or tool extensively in my current work. For instance, I am highly skilled in C#, which I can quickly refresh and use effectively when needed.

Currently, I primarily use Python for most of my work, and I consider myself highly skilled in it, proficient in handling various scenarios and utilizing advanced features. However, I acknowledge that Python has limitations, particularly in terms of memory management and the Global Interpreter Lock (GIL). When I need to write highly efficient or parallel code from scratch without relying on third-party libraries, I may need to conduct additional research and training.

On the other hand, when I say I am 'familiar with' certain tools, it means that I have used them for specific tasks, such as LLVM for static analysis or KLEE for test input generation. While I may have made minor customizations or adjustments for my needs, my familiarity with these tools does not imply an in-depth mastery. Nonetheless, I have a strong interest in expanding my knowledge in these areas and plan to do so in the future.

8. Why did you choose X instead of Y, which might have been more promising (e.g., exploring techniques beyond mutation testing, considering other tools, etc.)? Do you think you made a mistake in your decision?

As a novice researcher, predicting the best path was challenging, and it's quite possible that I made some less-than-optimal choices.

9. Are you solely interested in mutation testing and machine learning?

While my initial research focus was centered on applied machine learning within the context of software testing, I want to emphasize that this does not define my exclusive research interests or primary focus. It represents just one chapter in my broader research journey, which continues to evolve and expand.

10. You need to write papers during your study, but your writing TOEFL score isn’t high. Do you have any opinion on how you can deal with this issue?

I embarked on the journey of learning English on my own, focusing on the skills I felt were most necessary at the time. Writing was one of the later skills I tackled. While I acknowledge that my writing, among others, may not be perfect, I believe it's essential to take TOEFL scores with a grain of salt as they may not fully reflect a candidate's abilities.

TOEFL does not necessarily simulate the process of writing academic papers comprehensively, and it's susceptible to various factors, including external pressure to prioritize TOEFL preparation over actual writing proficiency. I am aware of the challenges I face, and I am actively working to improve my English skills.

If you have concerns about my writing abilities, I encourage you to review my paper titled 'FrMi: Fault-revealing Mutant Identification using killability severity.' This paper represents one of my earliest English writing endeavors, and it was completed before tools like ChatGPT were available. I made efforts to ensure its grammatical correctness using Grammarly. Since this paper was one of my first writing experiences, it can provide

you with a worst-case scenario for your assessment, and I am confident in my ability to improve and excel in academic writing as I continue my studies.

11. Your writing TOEFL score is okay, but it might not reflect your scientific skills. Can you provide a way to assess your writing skills for this purpose?

Indeed, a TOEFL score may not fully capture one's scientific writing abilities. To gain a more comprehensive understanding of my writing skills in the context of scientific research, I kindly refer you to my response to the previous question: 'You need to write papers during your study, but your writing TOEFL score isn’t high. Do you have any opinion on how you can deal with this issue?'