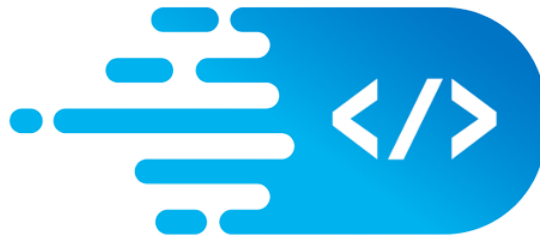


[IPRO 497-622] Startup Launch: Venture Design



AlgoPi

Team Members

Member	Role	Contact
Sofia Yang	Project Manager	syang78@hawk.iit.edu
Zuhuang Wang	Event Coordinator	zwang235@hawk.iit.edu
Zain Farhat	Data Analyst	zfarhat@hawk.iit.edu
Chris Hunter	Admin	chunter3@hawk.iit.edu

Introduction

AlgoPi is dedicated to helping students and aspiring software developers succeed in their career. We recognize the gap between the skillsets of junior/senior undergraduate students, and the role requirements in the software engineering industry.

Our team's purpose is to help students bridge the gap between college-level curriculum and the skills needed to find and succeed at their first career position as a professional software engineer/developer.

Problem Statement

“How can we help IIT CS students develop their **problem solving & analysis skills** to gain their first career position as a **software engineer**?”

Many CS students struggle with finding an entry-level position as a software engineer after they graduate, and feel that they are lacking the skills to succeed at their career.

These students would like to find opportunities to strengthen the necessary problem-solving and communication skills so that they can find and excel at their first early-career position in the industry.

Analysis / Proof of Demand

Our team collected data through a set of 20 in-person and online interviews with CS, and ITM (information technology management) undergraduate third and fourth year students, as well as industry professionals (software developers and hiring managers).

The students were asked about their previous experiences with applying for software engineering positions, and which aspects they struggled with or found the most difficult. Then the students were asked about how they prepared or attempted to handle those difficult aspects of job seeking and interviews and the results of those attempts.

The results were tabulated, and the data below shows the categories that students struggle most with when applying / interviewing for positions.

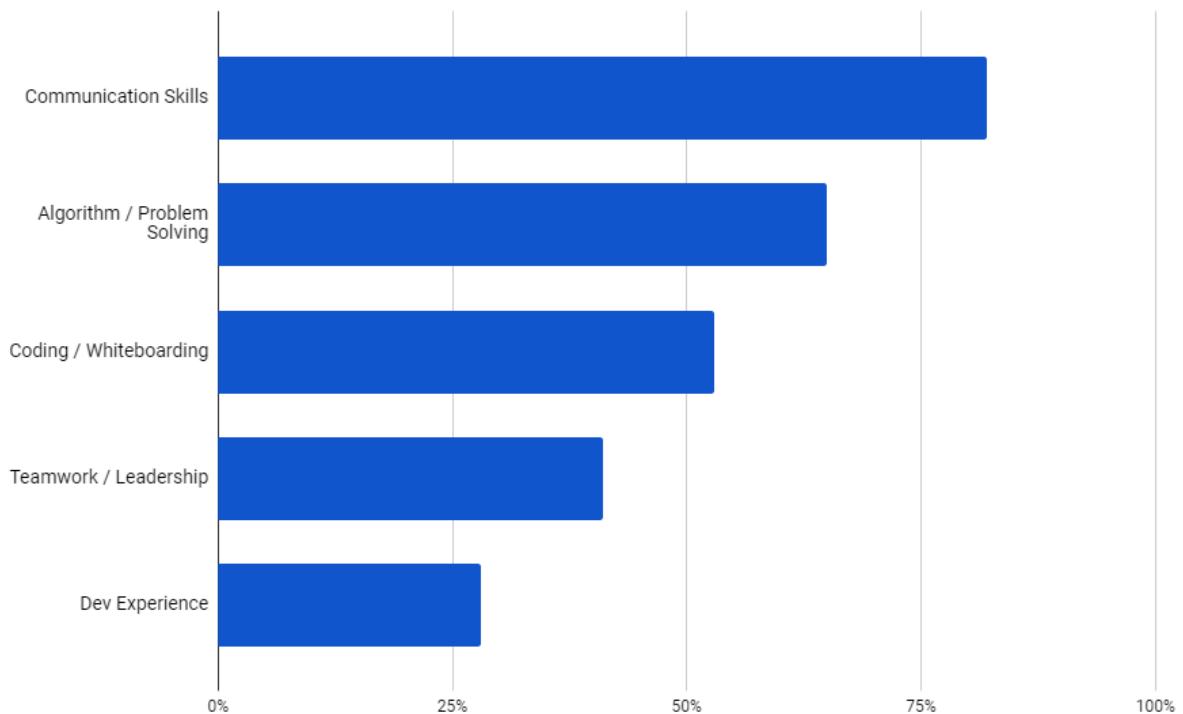


Table 1: Student interview results.

Over 75% of participants stated that they struggle most with clearly communicating during interviews, and over 60% responded that they struggled with algorithm/coding problems during a technical interview. More specifically, most of those participants struggled with coding and 'whiteboarding' (sketching out a solution pseudocode on paper / a whiteboard).

The team also had conversations with a few industry professionals, who were asked about their experience on gaining their first position as a software engineer and their advice for students. The industry professionals were also asked what they found most important when viewing a job application and interviewing students.

Insights:

- For early career positions, hiring managers value problem solving and communication skills over actual development experience.
- The purpose of technical interviews is to evaluate an applicant's thought process and problem solving skills, rather than finding a solution to a task in the shortest amount of time.
- Communication is one of the most important skills that employers look for – this includes writing clean and readable code.

Solution

The team's solution is to develop an integrated platform for users to develop algorithmic and problem solving skills through multiple sources, including online coding challenges, learning resources, and hands-on events tied into real-world applications.

The goal of this solution is to let students develop a methodical procedure for solving problems in a teamwork environment, and communicating their solution. In this way, they can build up skills that they can demonstrate when applying for software development positions, and use throughout their careers.

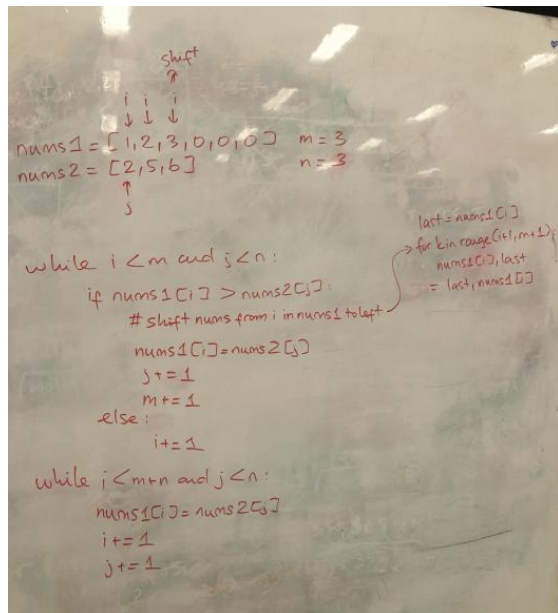
Due to limited resources and time constraints, the team prototyped and tested various aspects of the solution, as shown below:

Prototypes and Testing

Various aspects of the team's solution were prototyped and tested within AlgoPi's Discord community.

The first prototype was to select weekly coding problems of different difficulties, and have the participants solve the tasks and explain their solutions through code comments and written descriptions. The experiment was tested for five weeks and achieved positive feedback from the community. Overall, there were over 100 solution submissions from more than 15 participants. Over 70% of the submissions included solution explanations.

Examples of some solutions the participants wrote are shown below:

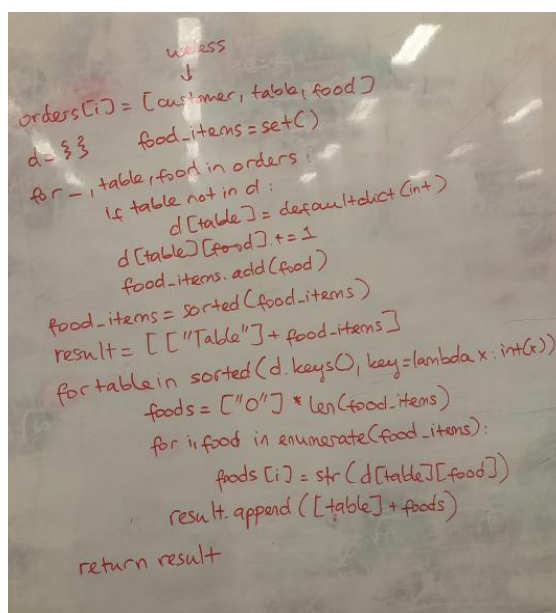


```
shift
↓ ↓ ↓
nums1 = [1, 2, 3, 0, 0, 0] m = 3
nums2 = [2, 5, 6] n = 3
↑
j

while i < m and j < n:
    if nums1[i] > nums2[j]:
        # shift nums from i in nums1 to left
        nums1[i] = nums2[j]
        j += 1
        m += 1
    else:
        i += 1

while i < m+n and j < n:
    nums1[i] = nums2[j]
    i += 1
    j += 1

last = nums1[i]
for k in range(i+1, m+1):
    nums1[k], last = last, nums1[k]
```



```
useless
↓
orders[i] = [customer, table, food]
d = {} food_items = set()
for i, table, food in orders:
    if table not in d:
        d[table] = defaultdict(int)
        d[table][food] += 1
        food_items.add(food)
    food_items = sorted(food_items)
    result = [ [ "Table" ] + food_items ]
    for table in sorted(d.keys(), key=lambda x: int(x)):
        foods = ["0"] * len(food_items)
        for i, food in enumerate(food_items):
            foods[i] = str(d[table][food])
        result.append([table] + foods)
return result
```

1. Split array first N elements --> 'first half'.
Last N elements --> 'second half'.
 2. Find all possible subsets a of the first half.
 3. Find and record sum_of_values() for all subsets b of the second half.
Sort the recorded sum_of_values() array.
 4. Loop through all subsets a, and binary search for the sum_of_values() closest to ((sum_of_values(ARR)/2) - sum(A)).
 5. Record the minimum of sum(A)+sum(B) - (sum_of_values(ARR)/2).
- To find all subsets of each half, we make a binary number of N bits, and use 1 for each bit to represent if it is part of the subset, 0 if it is not. For N = 15, this means 2^{15} possible subsets.
- 🤔 4

The second experiment was composed of hands-on ‘whiteboarding’ events, to simulate a real technical interview, with added teamwork components.

Coding problems are given to different teams of participants. The members of each team work together to brainstorm solutions, and explain their solutions either verbally or through written explanations. The team members then ‘peer-review’ each other’s solutions based on the solution efficiency, code clarity, and quality of explanation.

Due to limited resources, the team held these events virtually.

Due to the team’s lack of experience in hosting events, the participation numbers were low, even though the satisfaction rates of the participants were high.

The team then conducted a post-event analysis by sampling and interviewing 15 members of the Discord community. It was found that 73% were unable to attend due to scheduling conflicts, or weren’t aware of the event happening. About 30% of the people polled didn’t join because they were unsure of the event’s difficulty level. However, 66% of participants stated that they would be willing to participate in events in the future.

Some future improvements to this ‘prototype’ would include improving the team’s event organization skills, and either hosting events at more time slots, or holding more hybrid/asynchronous events. Other feedback gathered suggests that a future improvement would be to offer events for participants of different skill sets / skill levels.

Conclusion

AlgoPi will continue to grow and expand the solution while incorporating feedback from users. At the current stage, the team has identified a critical problem and audience, as well as developed and prototyped aspects of a solution. The results of prototype testing suggest that there is a demand for the solution, aside from the team’s lack of experience with solution execution. The future of our team will be focused on fine tuning the existing prototypes to better suit users’ needs, and then develop new aspects of the solution.