Gabriella Barcinas
UWP 001
Professor Hay
30 November 2020

Computer Science as a Discourse Community

Abstract

Computer science is one of the crucial career paths that an undergraduate student can major in, but there tends to be a lack of knowledge of what computer science truly is. The computer science major is usually seen as a difficult and impossible area to major in and this instant association discourages many individuals to consider the opportunities that it provides and its importance in societal advancements. To alleviate the common misconceptions of the major as one that is merely "hard," I have conducted two interviews with individuals involved in the field and the major to articulate how members in this community at UC Davis operate. Supporting evidence for my interviews are drawn from the UC Davis Computer Science Department website and the book "Discovering Computer Science" by Jessen Havil. From my research, it is clear that students depend on communication through online forums which enables them to solve problems through creating efficient software that can be applicable in society.

Introduction

The field of computer science is a particularly young discipline but continues to evolve over time. For being such a new field, the amount of knowledge concerning computer science and the work that it entails certainly falls short to what most students know about the major and the job professions it can lead to. Consequently, female representation in the major across universities and colleges, as well as in the field, remains scarce in comparison to males which results in a distinguishable gender gap. As a female undergraduate in the process of switching to this major, my goal is to provide more insight into the major of computer science that is offered at UC Davis for prospective female undergraduates, and underline how it follows the characteristics of a discourse community that John Swales defines in his book "The Concept of Discourse Community". According to John Swales, a discourse community is one that surrounds a common goal with an ongoing channel of communication amongst its members (221). A significant component in understanding the major of computer science is through analyzing these forms of communication between students as developing scientists and how it corresponds to the major's overall goals. The emphasis of the major is to provide students the tools to think critically when solving problems in software engineering in order to progress human-to-computer interaction as the role of technology continues to grow in our everyday lives. Considering the unique communication methods in the major will aid the understanding of how the major functions to achieve such goals.

Methods

To offer guidance in the major of computer science at UC Davis, interviews were conducted with a graduate student, Tao Wang, from UC Davis and an alumni from UC Irvine, Ozleene Osorio, who is currently a software developer at Etsy. The interviews will be used to analyze the main goals and communication methods of the major while offering insight into first-hand field experience. To support the findings in the interviews, evidence drawn from several scholarly sources will also appear in the research of this essay. One of the scholarly sources that will be referred to is the book "Discovering Computer Science" by Jessen Havil, an active researcher and professor at Denison University, providing information concerning the realm of computation and its function. To further the understanding of potential career prospects of the major, the UC Davis computer science department website will also be used to show the plethora of opportunities in computer science and the requirements of joining this community. A genre analysis of a common question-and-answer website used by computer scientists called GitHub will reveal the methods of communication that takes place in this discourse community as well. GitHub is a collaborative channel that is utilized in both the major and field of computer science and will show how computer scientists find solutions to the various problems they are dealt with. To integrate this collection of research with one another, the major of computer science will be examined as a discourse community built upon John Swales', a scholarly professor of linguistics, book "The Concept of a Discourse Community". From Swales, the defining factors of a discourse community is dependent on six criteria, four of which will be reviewed in this essay: shared goals, specific lexis, membership criteria and expertise, and genres of intercommunication.

Results and Discussion

Shared Goals and Interests

Firstly, the curriculum of computer science is vast and discoveries continue to emerge in the tech community. From Swales, a discourse community can be identified based on the shared goals and interests that are exercised by each member of the community (220). Students in this major will form skills in programming and computation to address problems inherent in the real world. In layman's terms, computation involves a series of steps that results in a "desired solution" for a specific problem which considers an input followed by an output. Essentially, students will attain critical thinking skills to execute such complex problems evident in all courses offered in the computer science major.

In my interview with graduate student Tao Wang, a teaching assistant in the introductory programming course (ECS 32A) at UC Davis, Tao was able to explain the various goals that students within the major prioritize. It is evident that computer scientists "try to solve complex problems" using the physical components of computers that perform based on a softwares set of

instructions (Wang). Similarly, software engineer Ozleene Osorio states how computer scientists seek "to understand how [they] can use computers to make life easier and efficient." Foremost, the UC Davis computer science department website, in the College of Letters and Science, outlines that students acquire necessary skills in software with a minor emphasis in hardware skills. The problems that computer scientists deal with can relate to almost any job where technology is involved; on a large scale, creating artificial intelligence for human use and on a more minute scale, creating the interface for social media outlets that allow users to communicate with one another (Wang). Wang also mentions the extreme importance of math within the major. Wang highlights how "most people around [him] have a mathematics background" and a main reason why students join this community is to pursue math to a certain degree, "but did not want to do pure math in college". Overall, the major of computer science at UC Davis teaches students to think critically in order to create, progress, and enhance systems in computer technology and prepare students to perform real world applications in multiple disciplines.

Specific Lexis

In order for students in the computer science major to reach such goals outlined above, students will need to gain the necessary knowledge to perform computation which involves a complex understanding of programming terms and their functions. In a discourse community, communication is accomplished using an established set of "lexical items" that is primarily understood only by its members (Swales 22). In this community, communication between students occurs with the use of programming terminology. Course learning in this major involves a specific comprehension of software which revolves around algorithms and is a distinct component in programming languages and allows humans to clearly communicate with computers (Havil 4). One can see a programming language as a medium that allows computers to interpret our English language into a form that a machine can understand. Python, JavaScript, and C++ are but a few of the programming languages that a student may come across and are based on the interior of a computer (Havil 20). Algorithms make up the many operations that scientists use to create programs that align with the problem they are trying to solve (see Figure 1 and 2). For a gage on the wide range terminology that students will acquire in this field, software engineer Ozleene Osorio provided some insight towards her experiences with computation. When asked the most important/used operations when developing a program, Osorio states that it is important for a scientist to "know their data structures and how/when to use one." She then follows up by giving an example of a data structure called an "Array," which is "a list that holds data," and the "most used operation on an Array is iterating through an Array" using a loop. Looping is one of the many operations that students will need to understand in this major for basic programming skills. Swales explicitly states that "information technology discourse communities" obtain a range of terminology in order to achieve their collective goals, which in this case is knowing the wide range of operations necessary to form a program, dependent on the computer language one is using (222). The terms "algorithm," "array," "loop," "data structure,"

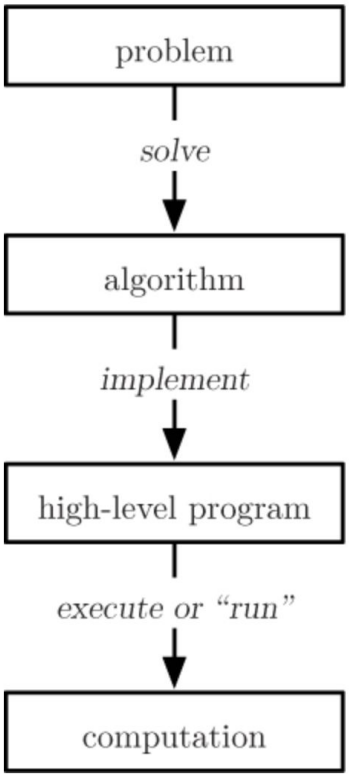
are some of the foundational terminology that students will use religiously to communicate with their peers and understand programs. This community relies on the common knowledge of a shared language needed in coding/programming that allows for efficient problem solving.

Figure 1: Example execution of an algorithm on sphere volume in the language Python from the book "Discovering Computer Science" by Jessen Havil

```
def sphereVolume(radius):
    volume = (4 / 3) * 3.14159 * (radius ** 3)
    return volume

result = sphereVolume(10)
print(result)
```

Figure 2: The diagram above shows the components that lead to a computation drawn from Jessen Havil's book, "Discovering Computer Science"



Membership Criteria and Expertise

In addition to the learning material that stems from the goals of computer science, there also lies specific criteria that a prospective member must meet before they can fully join this discourse community (Swales 222). The UC Davis computer science department website clearly outlines the necessary steps in declaring this major and the requirements to graduate with a Bachelor's of Science. It is important to note that for anyone planning to switch majors while currently enrolled as a student, they must finish at least one quarter at UC Davis. First and foremost, a student must achieve a C- or better in the required set of preparatory courses and have at least a 3.0 overall grade point average at the time of petitioning to switch to computer science. In the preparatory subject matter, students will attain background knowledge of mathematics, computer science, and natural science before diving into more complex material. Some of the courses that students will need to take are: calculus, specifically the MAT 21 series, discrete mathematics for computer science, ECS 20, programming and problem solving, ECS 036A, and software development & object-oriented programming in C++, ECS 036B. While in the major, students will center their focus around computer science engineering such as designing algorithms, computer hardware, probability theory, and of course, intricate programming languages. Students also are given the ability to choose from a set of electives which enables one to branch out into different disciplines of computer science and find what best suits their interests and skills. Despite the vast course requirements, when interviewing Ozleene Osorio upon the qualifications of being in this field, she says that "any student can join the computer science major as long as they have the interest and drive to want to pursue it...I, personally, didn't have any background in computer science." A driving factor of joining this community is the commitment to learn and having the drive to do so. As mentioned before, this major and field strives to recruit individuals in STEM (science, technology, engineering, and mathematics), especially women, since the need for computer experts is ever-growing and is a contributing factor to societal advancements.

Additionally, like most majors and fields, experience can lead to greater opportunities and is commonly associated with someone's level of expertise. Ultimately, the "survival of the community depends on a reasonable ratio between novices and experts" where each level of expertise is significant to the progression of computer science (Swales 222). This major strives to develop well-rounded problem solvers in an array of specialized areas who will hopefully impact the world post-college. Ozleene Osorio describes how "students who graduate from this major usually become Software Engineers and work for a tech company" while a Chief Technology Officer or a Distinguished Engineer are considered higher leveled positions in the field. Each specialized area is different and may require more skills in that particular realm. For example, Data Scientists and Machine Learning Engineers are other disciplines that "require more knowledge in mathematics and statistics" (Tao Wang). Overall, there is so much to learn in and outside of the major, and on top of that, one must have the ability to adapt to the latest technology trends in order to be successful.

Genre Analysis

Though the goals and requirements remain significant in understanding the major of computer science, online forums are a specific medium that allows communication to take place in this community. There are many platforms that computer scientists utilize in order to collaborate with one another for help, guidance, peer-review, etc., in programming specifically. Computation can be very complex, and programs continue to be altered to improve functionality and efficiency. Based on Swales, "a discourse community utilizes and hence possesses one or more genres in the communicative furtherance of its aims," which is shown in the methods of intercommunication by students through the website GitHub (221).

Github is one of the many code hosting platforms that computer scientists utilize to collaborate and communicate with one another. The audience in this particular forum are software developers who are students, or even professionals in the field, seeking knowledge and guidance in computation from other scientists like themselves. Interviews Tao Wang and Ozzleene Osorio mention how the website attracts both students who are working on team projects for a computer science course and professionals who are building projects for their company that are eventually applied in the real world. Again, developing code is one of the important roles of a computer scientist and GitHub encourages collaboration to form programs successfully. GitHub is highly known for "version control," which is the management of information such as computer programs and tracking changes in code. The most important communicative features of Github are: "opening issues," a discussion thread where members can communicate directly with one another about their projects and ask questions; "creating branches," where changes of a code are made separate of the original code by team members; and forming "pull requests," which are proposed changes by a member that the whole team can review and communicate upon its accuracy. Additionally, the website allows all software developers to share their projects for the public use and spread useful knowledge to other scientists in the "Explore" tab where one can benefit from the research and trial-and-error processes of other scientists.

The communication in this genre is performed with an objective in mind, so the tone is merely factual. Users of GitHub must have a background in software development in order to participate as a collaborator and have experience with the project issue as well. This site is specifically made to bring all members of the community together to communicate and collaborate through a versatile medium. Working with a team is a common element in computer science careers, and GitHub provides the means for such necessary communication between students to execute precise software for solving problems (Figure 3 shows the layout of GitHub from a users perspective).

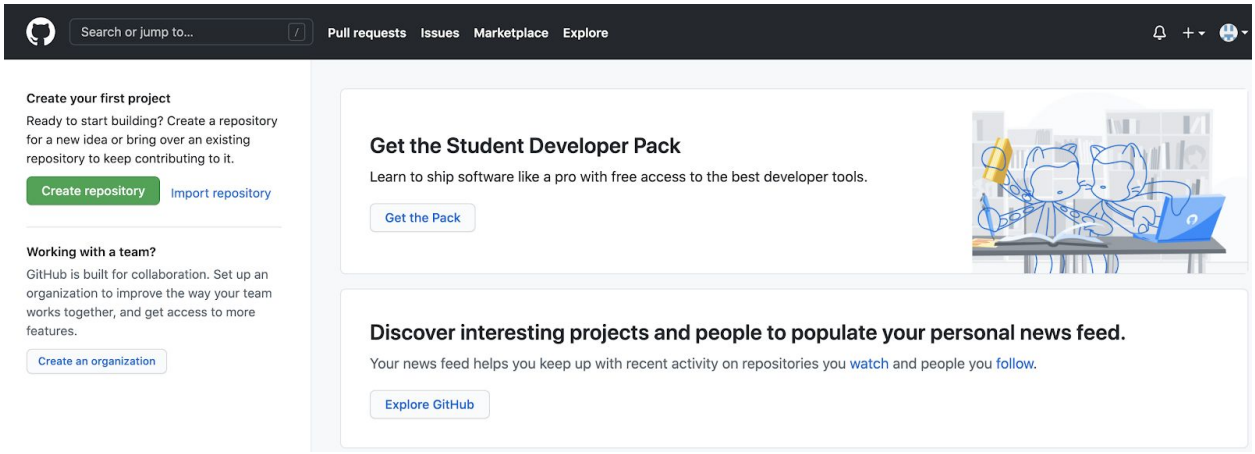


Figure 3: Overview of the GitHub website from the home page

Conclusion

Based on my research and interviews with experienced computer scientists Tao Wang and Ozleene Osorio, the values of this community offers a better understanding of what it means to be in this major and the steps that are needed to be successful. Discovering the precise goals of the community, by analyzing the mechanisms of communication between its members, reveals how important collaboration is when creating applicable software and finding solutions to a problem. This discourse community aims to develop scientists who can take on problems in the real world and have the ability to maintain information systems that are implemented throughout society. As the role of technology continues to increase, so does the demand for experts in the field and the need for women in STEM as well. In order to prepare students for these career destinations in different branches of computer science, students will take foundational courses in computation leading to more complex material in software engineering with the ability to choose elective courses to help find one's strengths and passions. On a large scale, many other fields rely on computation which demonstrates the professional relevance of the major. This research provides a broader perspective of the possible career destinations that result from a computer science degree and hopefully enlightens young female undergraduates of the opportunities and real-world outcomes of joining this discourse community.

References

Barcinas, Gabriella, and Ozleene Osorio. 3 Nov. 2020.

Barcinas, Gabriella, and Tao Wang. 10 Nov. 2020.

Computer Science Major at UC Davis. 19 Feb. 2019, www.ucdavis.edu/majors/computer-science/.

"What Is Computation?" *Discovering Computer Science*, by Jessen Havill, CRC Press, 2015, pp. 1–5, learning.oreilly.com/library/view/discovering-computer-science/9781482254167/.

Swales, John. "The Concept of Discourse Community." Genre Analysis: English in Academic and Research Settings. Boston: Cambridge UP, 1990. 21-32. Print.