Ryan Yerge

COP 1500 · 3/28/19

Out of the numerous academic disciplines offered at higher learning institutions, the computer science discipline has garnered some warranted attention in recent years. In an article published by Liana Loewus for Education Weekly, the number of computer science graduates has risen sharply since experiencing a decline between 2003 and 2009 (“Computer Science”). The realm of computer science is a continuously expanding field in the study of computing, ranging from hardware development to software engineering, and it continues to innovate and grow as time progresses. It is comprised of a multitude of diverse topics founded on discrete mathematical concepts such as Boolean logic, while also relying on engineering principles for software development and information technology systems. At times it can be daunting for some to discern between the various branches of computing. For the sake of this essay, the focus will remain within the computer science, information technology and software engineering disciplines.

Although the disciplines contain very similar curricula, it is important to distinguish between them respectively. Computer science rests at the helm in which the breadth spans all the fundamental concepts of computing theory like algorithm analysis and other discrete structures. Its far reach includes, but not limited to, designing software, computer architectures, programming languages, and optimization of networks. Information technology is often an overlooked discipline, but it is a vital component to an overwhelming amount of organizations consisting of medical facilities, government agencies, and educational institutions. According to “Computing Curricula 2005,” information technology graduates’ specialty lies in the name: *technology* that manages *information*. IT professionals are responsible for selecting and implementing products within an organization to optimize the organization’s performance. Software engineering shares the most overlap with computer science programs. Software engineers are the creative minds behind the design and implementation of software systems and applications. From a profession standpoint, software engineers are tasked with receiving a client request for the ideal software catered to their needs, and then the request is translated into a developed software program that meets the client’s expectations (p.13).

At its core, computer science can be discussed within three fields: algorithms and data structures, cybersecurity, and hardware. Data structures require unique architecture and design to store and manage loads of data, and tandem are recursive algorithms to search and fetch data. Computer scientists are constantly looking for new ways to develop more proficient algorithm programs to navigate data structures in a search engine. While data structures can manage inventory for an online store, sometimes they are designed to store sensitive materials like personal financial information or health records. This is where cybersecurity plays its role in implementing protocols to deter any inadvertent or intentional intrusions where private information can be compromised. Additionally, the world of computing would be naught without hardware to power and direct instructions to the machine. The main driving force is controlled by the flow of electric current through millions, or even billions, of transistors establishing logic gates to help translate user instructions into the machine dialect of binary. Computer engineers research and develop ways to expand hardware whilst keeping the scale of the machine reasonable.

While all the fields of computer science are necessary and complementary, the one that stands out the most to me are algorithms and data structures. From simple arithmetic to building programs to model natural phenomena, some of the world's most formidable programming challenges need superior algorithms. Adjacent to recursive thinking, the data structures in which algorithms access information require clever design and thoughtful architecture. Assuming proficiency in this field will only serve me well while striving to become a software engineer that is second to none (Dale and Lewis 26-27).

Works Cited

Association of Computing Machinery. *Computing Curricula 2005*.

ACM and IEEE, 2006.

Dale, Nell, and John Lewis. *Computer Science Illuminated*.

Cathy L. Esperti, 2006.

Loewus, Liana. “Computer Science Degrees Hit New Peak.”

Education Weekly, 4 Apr. 2017, http://blogs.edweek.org/edweek/curriculum/2017/04/computer_science_numbers_degrees_new_peak.html

Short Report

When I began my research for this assignment, I first started by structuring my report on a sheet of paper. The crucial components of the prompt were outlined within the body of the report, and of course introduction and conclusion modules were included at the header and footer of my outline respectively. This portion of the process iterates the critical thinking FGCUScholars skill by offering a roadmap as to how the report will be presented and laid out for the audience. My next step was to begin researching the available resources listed on Canvas to find information relevant to the prompt. Eventually, I turned to benchmark organizations within the computing community like the Association for Computing Machinery (ACM) to search for clear and concise definitions of various computer science disciplines. I wanted to ensure the audience was receiving information that was not only reliable but revered in the realm of computer science academia as hallmarks of any computing curriculum. This showcases the next Scholar skill of information literacy in navigating the plethora of computer science-related information on the web and having the capability to discern opinionated rhetoric from accredited dialogue. Once all the content was collected, I was ready to begin writing. Writing is arguably one of the more important skills because it consists of translating all the researcher’s findings into a conveyable format for the assumed reader. The actual written text engages critical thinking again by forcing the author to question the information being conveyed. What is the information being presented? Why is it important? Or how is it relevant to the overall prompt? Asking these questions ensures that the overall message is not being lost or strewn through tangents of facts and information with nothing holding them together.