To: Engineering Communications  
From: Patrick Austin (10-1)  
Date: May 3, 2017  
Subject: Core Capstone Defense  
  
**Applying Knowledge**  
Patrick has applied his knowledge of mathematics in the design of circuits for use with Arduino microcontrollers. Creating safe and correct Arduino designs required calculations using the equation of Ohm’s Law to assure electrical inputs to the board were being resisted to a safe extent. Incorporating transistors and LEDs into Arduino designs required algebra procedures and calculations based on managing the voltage and flow of current into those components.  
  
In order to create complex programs Patrick has needed to understand and apply scientific concepts. The concepts Patrick has understood and utilized include principles such as abstraction and modularity, theories such as Turing’s thesis on computability, and the laws of time complexity, which govern the performance of a program as its input size approaches infinity.

In order to use these concepts in his programming, Patrick has understood and applied engineering concepts. He has applied laws of time complexity in order to understand the mathematical bounds of performance for common operations such as searching for an entry in a list. He has iterated repeatedly on programming projects, in order to optimize performance and expand functionality. He has used the engineering design process to guide this iteration by brainstorming, prototyping, and testing the correctness of his programs.

**Experiments and Data**

Patrick designed a computer simulation of possible ways to organize lines in which customers wait to receive service at a business place during the Computer Science 301 Data Structures class at the University of Nevada, Reno. Patrick followed the seven-step process for designing an experiment to test the correctness of the program.

The hypothesis was whether or not the simulation accurately simulated a line. The relevant variables were statistics about the operation of the line gathered during the simulation, like average wait time, average line length, and maximum line length. The level of treatment was confined to Patrick’s program and a set of small inputs, consisting of times of customer arrivals, where correct line behavior could be verified mathematically. Patrick controlled environmental and extraneous factors by limiting the number of computers and operating systems on which the program would be tested. Patrick designed the experiment to test the hypothesis by making the simulation output the computed statistics to a file, which was then compared against the correct values, which had been verified mathematically. This was done for each different type of line simulation and input file, effectively selecting and assigning subjects to groups. Patrick then pilot-tested and revised the structure of the statistic-gathering components of the program numerous times to improve correctness before conducting the final test. Finally, Patrick statistically analyzed the data gathered from the experiment on his simulation and compared it to the expected values which had been previously calculated.

Because of Patrick’s experimental design, he was able to generate high-quality data which could be analyzed in order to evaluate the hypothesis. This data was high-quality since the experimental design process had guaranteed it would completely, accurately, and consistently quantify the operations of the simulation. In this case analysis consisted of verifying a match between the measured data, the line performance statistics such as average line length, and the expected data. Patrick was able to analyze any errors that arose in the data by identifying systematic errors caused by bugs in the code as well as measurement errors caused by rounding and the finite precision of variables in the simulation. The experimental values from the program were compared against the expected values in order to evaluate the hypothesis, whether or not the simulation was correctly emulating the behavior of the line.  
  
**Contemporary Issues**Patrick has researched and explored an important topic in contemporary computer science in Engineering 301: the implementation of digital rights management in software. Digital rights management, or DRM, is used in the software industry to deter piracy by attempting to restrict usage of software to licensed users only. However, the choice of strategy used to implement DRM comes with costs and tradeoffs, ranging from performance overhead to the risk of unintentionally inconveniencing legitimate users or even locking them out from access to their software. Issues of consumer rights and protections in the digital space, the legal limits of current digital copyright laws, and the existence of a global culture of software piracy all come into play in this issue.

Patrick is also knowledgeable in another area of contemporary computer science: he has studied the network security risk of denial-of-service attacks. These attacks work by flooding a computer connected to the internet with a massive volume of network protocol requests which can have severe consequences for the public by inhibiting computer systems vital to the public’s health and welfare, such as those of the medical, energy, and banking industries. For example, a hospital unable to access patient records due to such an attack could be ill-equipped in a medical crisis. Patrick has studied precautions network engineers can take to limit the impact of these attacks, as well as the continued evolution of new denial-of-service attacks over time.

**Engineering Impact**

Patrick has received a broad education, including a past degree in political science, and has had broad experiences ranging from IT work for the university, to grading papers and tutoring students, to work in Reno’s gaming industry.

Patrick has studied the problem of digital rights management, or DRM, in the software industry. By the account of one industry advocacy group, software piracy accounts for two out of every five software products installed on computers worldwide, greatly impacting industry profits. DRM is used to deter piracy, but studies of consumer behavior have suggested that flawed and onerous DRM schemes can drive away more sales than they protect. Patrick has proposed an engineering solution: increased experimentation with and promulgation of DRM-free software.

On the global scale, piracy rates vary greatly from nation to nation, being generally higher in the developing world. For that reason, this solution might be a better fit for businesses in the developed world, where consumers have demonstrated some willingness to pay for the convenience of DRM-free software, and have less impact in the developing world, where rates of piracy are very high already and a DRM-free option might be ignored or misused to enable further piracy. Economically, this solution could lead to increased profits by generating increased software sales, but it is untested in many software fields and therefore could present an economic risk for software businesses to attempt, since promulgating DRM-free software can make piracy easier. Environmentally, this solution would allow for the elimination of DRM authentication servers, reducing the global carbon footprint; however, increased promulgation of software could lead to more computer usage by software consumers, which would have a mitigating effect. On a social level, DRM-free software could promote a culture of user-friendliness and consumer convenience in software design, but critics might fear that it would further promote and enable the existing culture of piracy by making software easier to steal.

**Multi-Disciplinary Team**

Patrick has worked, during his time as an IT Technician at the Extended Studies department of the University of Nevada, Reno, in a multidisciplinary team. This IT team consisted of fellow Computer Science and Engineering students, a student in the College of Liberal Arts, and graduates of Information Systems and Cybersecurity programs. Together, the team provided a well-rounded variety of disciplines and perspectives.  
  
This team used a variety of tools in order to function effectively. The team managed communication between multiple sites via the business chat program Slack, did scheduling via Microsoft SharePoint, took minutes of group meetings, and worked to maintain a database of documentation to address frequent issues and provide a guide to performing common tasks. For example, Patrick maintained documentation on the setup process to prepare a new PC for use in the computer labs maintained by the team, explaining necessary steps and best practices to coworkers from backgrounds other than Computer Science and Engineering.

The team was successful by several metrics. The team’s use of documentation reduced set-up time for new lab PCs by more than half. The team’s use of Slack to collaborate on solutions likewise reduced turnaround time for support tickets significantly, with 80% of all tickets being resolved in less than 30 minutes. The lab’s uptime increased under the team, to over 99.9%.  
  
**Professional and Ethical Responsibilities**Patrick recognizes that engineers are beholden to professional and ethical standards, and above all are obligated to hold paramount the public health, safety, and welfare. He acknowledges the need to put these standards into practice through strict adherence to the National Society of Professional Engineers’ Fundamental Canons of the Code of Ethics.  
  
Patrick has shown professional responsibility in the course of his employment as an IT technician at the University of Nevada, Reno. Patrick worked to provide capable, practiced, and skillful service to his employers at the Department of Extended Studies. He did so by applying his computer science and computer engineering training to handle day-to-day technical problems in the department. He also scrupulously followed legal standards involving the disclosure of personal and academic information related to students in the department computer labs, as pertained to the Family Educational Rights and Privacy Act.  
  
Patrick has exhibited ethical responsibility in his employment as a student grader for computer science courses at the University of Nevada, Reno. When asked by fellow students for advance information about upcoming material in the classes he was responsible for grading, Patrick declined to provide such material despite the possibility of selfish gain. In refusing, he applied Rule of Practice 1c. from the Code of Ethics because he refused to reveal facts, data, or information without the prior consent of his employer.   
  
**Effective Communication**Patrick has worked to improve his communication skills, and to use effective communication techniques in class and in his professional life. In his role as a teaching assistant for the course Computer Science 456 at the University of Nevada, Reno, Patrick was asked to facilitate study sessions for class examinations. The audience for these sessions was a group of approximately 25 seniors in the Computer Science department.  
  
Patrick employed a number of techniques and strategies to make these sessions successful. When taking questions from the students, he repeated the question to the whole audience, asked for clarification when necessary, and corresponded with students regarding questions he was not confident answering after the study session was over. When presenting material, he focused on providing concrete examples and moving through the class material at a comfortable pace.

This communication was effective. Patrick received verbal and email feedback from students praising the manner in which the material was presented. Additionally, Patrick was able to recognize where the students who had participated in the study session had made effective use of the format and style of his example solutions when he graded the subsequent examinations.  
  
**Life-Long Learning**As a student of computer science and engineering, Patrick recognizes the need to engage in lifelong learning. In order to be qualified and competitive in his field, Patrick must keep himself abreast of new technological and theoretical developments, in areas ranging from new developments in computer hardware, to new approaches and vulnerabilities in computer security, to new innovations in the design of programming languages. Patrick recognizes that computer science and engineering often requires certification on new topics to qualify for jobs and demonstrate competency in new and developing areas of the field.  
  
Patrick currently practices lifelong learning and intends to continue to do so. Patrick has increased his marketability and knowledge in the field beyond what his classes and work have required by learning the Java and Python programming languages in his spare time, which he uses for personal applet projects. Patrick also participates on online forums such as Stack Overflow discussing programming, where he keeps abreast of new developments by interacting with other students and professionals in the field. Patrick will continue to engage in lifelong learning by continuing the practices he has developed so far, as well as joining and engaging with the IEEE professional society, and subscribing to computer science trade journals such as the Communications of the Association for Computing Machinery.