This summer I carried out independent computer science research on the campus of the University of the South in Sewanee, TN. I studied under the supervision of Dr. Lucia Dale, and she advised the project based on her knowledge of my technical skill and particular interest. The internship was intended to give me a working knowledge of RSA and ECC cryptosystems (the details of which are examined below) and to improve my practical coding skills. I sent Dr. Dale any original code that pertained to the two cryptosystems, and supplemented my basic understanding with academic articles on the subject (the reading was added to take the place of code when the level of mathematics surpassed my ability).

The most tangible product of my work this summer is a program that encodes and decodes a user-entered message through RSA encryption. There are tools in the Java library that are pre-written by software developers to implement both RSA and ECC, but Dr. Dale and I decided to forgo using those tools and instead focus on developing my own. This was a decision between either producing sophisticated code using a tool that I would not fully understand because of my level of programming ability and producing simpler code that would force me to understand how each piece worked. While it would have been effortless to use the libraries already in Java, I am glad that Dr. Dale pushed me to be original and solve the problem for myself. It became immediately clear that working directly with RSA would be more feasible than working with ECC, as the mathematics surrounding ECC involve algebra on elliptic curves (a very difficult area in mathematics), while the security of data provided by RSA depends on the difficulty of factoring a product of two prime numbers.

My code for RSA encryption randomly selected two prime numbers, multiplied them
together, and then produced a public and private key for encryption and decryption (finding a public and private key is essentially a manipulation of the primes using modular arithmetic and the equation $\varphi(n) = n - (p+q-1)$, where $n$ is the product of two primes $p$ and $q$). Since Elliptic Curve Cryptography requires an understanding of mathematics beyond my ability, Dr. Dale and I edited the goal of the research. I was able to complete the first step in ECC: selecting a random point on an elliptic curve. This involved writing a program that would randomly generate the constants for the elliptic curve equation ($y^2 = x^3 + ax + b$, where $a$ and $b$ are constants), draw the curve, and select a point on that curve. From the relative difficulty of producing code for both systems, it was apparent that ECC provides the same data security as RSA with a shorter key (providing the benefit of less memory usage), but RSA has the advantage of a succinct algorithm.

In terms of skill development, this summer has helped me understand my strengths and weaknesses in computer science, and how I can improve my programming ability. My strength is examining and understanding the “big picture” of a problem, and my weakness is a tendency to become entrenched in the details once I begin. Moving toward a career in computer science, I think I would be well served by practicing simple problems as efficiently as I can, rather than focusing on the planning stage, and becoming more willing to discard code that reveals itself to be inefficient.

The great gift of this research was, unsurprisingly, the wisdom of Dr. Dale. Learning about cryptography was very interesting and enjoyable, but entering the computer science courses next year and then a job search with the advice and lessons of Dr. Dale will be
invaluable. Young computer scientists are stereotyped as flighty, always looking for the best editor or language or prewritten code to solve the problem, but Dr. Dale exemplifies a steady alternative. She encouraged me to use an editor without tools that would fill code in for me or indicate that a piece was incorrect and to use Java classes only when writing the code myself would be busywork. She has given me a solid foundation for the rest of my class work and career, and I am very grateful.

The experience of working with Dr. Dale has shown me that I am better suited for a career in computer science than one in mathematics. My major at Sewanee is math and I have found the subject to be my favorite class (or set of classes) in any given semester, but computer science is, for lack of a better word, *fun*. The aspect of math that I have always enjoyed is solving a puzzle, and the problems to solve in computer science are intricate and interesting. This summer has solidified my conviction that, while I will finish my undergraduate work with a degree in math, my career should be in a branch of computer science. At the beginning of this summer, I was planning on taking the GRE and applying to graduate school. Now, I am preparing for a job search, and I am much happier with that course of action. This research opportunity with Dr. Dale has introduced me to the field of cryptography, improved my ability to think critically, advanced my coding skills, and provided me with clarity about my future career.