Teaching Portfolio for Mark S. Sherriff

Associate Professor – Non-Tenure Track (Teaching)
Department of Computer Science
University of Virginia

January 2016

Supplemental material can be found at:
http://www.cs.virginia.edu/~sherriff/

Personal Statement of Teaching for Mark S. Sherriff

As a teaching non-tenure track faculty (NTTF) member, my main focus has been undergraduate education. During my time at UVa, I have taught over fifty course sections, facilitated nearly 100 lab sections, managed around 200 teaching assistants, and taught over 5000 students – the most of any SEAS faculty member over the same time span. I thoroughly enjoy the challenge of teaching and am looking forward to many more exciting and productive years as a professor.

Teaching Philosophy and Methodology

My goal is to create learning experiences that engage the student both in and out of the classroom. Students should come to class, learn something new, work with their classmates to form a better picture of the concept in their mind, and then find places in their life where the ideas are real and apparent. I have found that in the field of computer science, even the most abstract theory can be made more concrete via real world examples and analogies.

In the years that I have been teaching at the University of Virginia, I have followed my methodology of real world examples, active learning, and project-based learning. From my 500-student introductory course to my relatively smaller 80-student upper level electives, I endeavor to make each class a unique learning experience, whether that means bringing in props to illustrate a point or creating a scavenger hunt that sends students all around grounds. I believe that any class that is stagnant in its presentation and involvement with students is missing a huge opportunity to engage and challenge students and to recruit them into computer science. Being able to create these experiences for new engineers is even more special, as I cherish every time a student tells me that they declared computer science as their major because of my class.

I follow the model of "Inform, Experience, Implement" in my teaching. This cyclical pattern first establishes the history or interest of a topic to ground it for the student in real life. Then I move into an activity where the student can experience how the problem or technology affects them. Finally, we transition the problem into computer science, teaching the algorithms and techniques used to solve the problem. I have used this model with middle and high school students and all levels at UVa with great success.

Teaching Innovations and Course Creation

Using my methodology of "Inform, Experience, Implement," real-world examples, and active learning, I have created some unique classroom experiences that have been successful and have been recognized for being innovative. When I was teaching CS 2110, I introduced an activity called the Encryption Chase. The lesson began with me entering the room and informing the class that their lecture for the day was encrypted and that they would have to figure out how to decrypt it in order to learn the material. It just so happens that the lesson that day was on various encryption methods. When students solved the first clue, it sent them to various other places around grounds to solve other clues. Each clue was encrypted with a different technique, forcing the students to learn as they went. Later, when I began teaching CS 1110, I adapted the Encryption Chase to teach looping constructs in Java. A basic Caesar cipher is simply a rotation of each character on a string. Students learned how to write loops while decrypting text I had hidden around grounds. The Encryption Chase was named one of the "Nifty Assignments" in CS in 2010 at the ACM SIGCSE conference.

Transitioning from my lower-division courses to the upper-division courses, I also incorporated new ideas into CS 3240 (Software Engineering). The first semester I taught the class marked the first time the department has used Lego Mindstorm NXT robots for the course projects. The Lego kits gave the students exposure to Bluetooth, basic mechanical engineering, TCP networking for communication, and physical data sensors (light, ultrasonic, touch, etc). While this project-based learning exercise is extremely important, I also focused on their role in the industrial workplace. All teams were required to give a formal and informal presentation to their classmates. Today, Lego Mindstorm robots have been purchased based on the work that I did in CS 3240 for continued use in the course in the future.

Another innovation that has worked well is the pairing of teaching mobile technology with teaching web services. By themselves, each concept could be the main topic in a separate course. However, I felt that showing how low power devices and devices that could disconnect and reconnect from the network at any time provided unique challenges for web developers. The course would provide students with the skills they need to succeed in an Internet-connected, mobile-wielding world. In support of this program, I received a gift grant from Google for 29 Android devices and another gift of Google TV devices. Students built intricate projects that balanced features between Internet-facing web services and Android-powered apps. Projects have spanned a wide range, from flash card apps for young students to apps that do optical character recognition. Students have thoroughly enjoyed the class and it was officially added to the UVa Course Catalog as CS 4720: Web and Mobile Systems. I have published my work on this course at the IEEE/ASEE FIE conference. Since then, the course has grown dramatically and we have obtained class sets of Nexus 7 tablets and iOS devices for teaching both mobile platforms to around 80 students a semester.

Also in the realm of project-based courses is the other new course that I have helped to create with Prof. Aaron Bloomfield. In the Spring 2012 semester, we taught the first instance of the Service Learning Practicum course, which gives teams of students the opportunity to work with actual customers by helping to solve the software needs of non-profit organizations in Charlottesville and beyond. The students are learning valuable real-world lessons about software development while also providing a great service for worthy organizations. My team built a family and home repair tracking system for Appalachia Service Project. The system went live in June 2012 and will be actively used across five states, aiding summer staff and volunteers with helping those in the greater Appalachia area. The course has now become the foundation of our capstone track for graduating seniors, and has generated numerous software solutions for non-profits in the Charlottesville area.

Teaching Evaluations and Accolades

During my years at UVa, both my students and my colleagues have recognized me as an effective teacher in SEAS. My overall teaching evaluation scores are around 4.5/5.0 across all of my courses. I consistently beat the SEAS average in every teaching evaluation category. While doing this, I have taught more student credit hours than any other professor in SEAS over the same time period (15924, with the next highest at 10484), and rank in the top 10 for all of UVa. My largest classes crest at around 500-600 students (CS 1110) and even my upper-level elective courses are routinely 70-80 students. Students report in my evaluations that I create a classroom experiences that students from both SEAS and the College enjoy and recommend to others. They also report that they learn a great deal from my class and many are motivated to continue their education in CS. It is always a wonderful feeling to be told that you helped someone find their academic passion in college. I am extremely proud of my teaching record at UVa.

For my efforts, I have been award the UVa All-University Teaching Award and the first ever Hartfield-Jefferson Scholars Teaching Prize, worth $12.5K. My nomination and selection came from both students and other faculty in SEAS. I have also been awarded the Trigon Engineering Honor Society's faculty award and have been named the ACM CS Professor of the Year twice.

Teaching Self-Improvement

One of my main vehicles for self-improvement and reflection comes from podcasting. I have podcasted every lecture from every class for the past six years. These recordings are incredibly valuable for revisiting my lectures and teaching style when I evolve a class from year to year. Along with my podcasting efforts for self-reflection, I have exercised other avenues to improve my teaching. I have attended teaching workshops offered by the Teaching Resource Center at UVa (although these workshops tend to be geared more toward small liberal arts courses). I have been attending the SIGCSE conference and have also served on the organizing committee for the CSEET (Conference on Software Engineering Education and Training). I am staying active in the CS Education community and have published papers in various conferences, while also helping to run tutorials and educational sessions at these conferences.

Teaching Conclusion

I feel that one of my greatest assets is my talent to teach. I am proud of my teaching evaluation scores over the years at UVa and the four years prior at NC State, especially in categories such as my enthusiasm for teaching, my course effectiveness, and whether the student would recommend me to other students. I am fortunate to have been able to learn from exceptional instructors at Wake Forest, NC State, and UVa, and I have tried to incorporate their various teaching styles into my own. I strongly believe in the strengths of problem-based learning, especially in a field such as computer science. I continually try to involve the class in every activity via active learning techniques, as opposed to having a strictly lecture-based class.

The computer science field is becoming much more diverse, with new students of various backgrounds and age groups. People are turning to computer-related fields for opportunities to start their career in a growing area, jump-start their current career, or just out of curiosity. Some students will come in ready to learn, while others want to get in and out as fast as possible so they can move on with their careers. These different approaches, backgrounds, and motivations create an interesting environment in which to teach. I try to teach my classes to reach all of the students in my class, not just those who are there for reasons that are more in line with my thinking. Through my experiences as an instructor, teaching assistant, and through my academic work, I am confident that I am an effective instructor in the field of computer science.

Teaching Responsibilities

One of my primary responsibilities as a teaching faculty is to teach the large, lab-based undergraduate courses. I have taught multiple sections of CS 1110 (Introduction to Programming) and CS 2110 (Software Development Methods). These classes work together to bring students with no programming background up to a level where they can create small- to moderate-sized software systems. I have worked closely with the other faculty in the CS department, particularly the other teaching faculty, to reimagine and improve these courses for a new web-centered, mobile world.

In addition to teaching the intro courses, I typically teach one upper-level elective each term, usually with around 65-70 students. The topics for these courses usually lie in a systems development area, whether the system is a group software project, a web/database application, or a mobile system. My upper-level courses focus on software development in the real world and project-based learning. As a software engineering researcher and practitioner, I want my students to have as much experience gathering requirements, creating software designs, and working together as a team as possible before entering the work force. I also give my project-based courses a great deal of flexibility in the nature of the system they build. Students are often more motivated when they have a vested interest in the system being created. I have seen a wide range of projects over the years, from TA application systems, to course scheduling aids, to flash card mobile apps for kids, to picture sharing websites. I am proud of the software my students have created as a part of my course.

In all my courses, I focus on active and example-based learning. My lectures are rarely typical lectures. I focus on small group exercises (even in classes over 250 students) where students break out with their neighbors to code a simple problem or design a particular algorithm. Sharing those designs and then arriving at a solution together is a core part of my typical classroom experience. I also try to get students moving as much as reasonably possible. One of my favorite activities is my scavenger hunt game that I use in as many classes as possible to get students out of the classroom itself to "discover" the lesson for the day.

My goal is to give students a reason to come to class. The recent discussions revolving "flipping the classroom" feels like something I have been emphasizing all along, even if I don't call it that or follow the model perfectly. My classes are about engaging the students and working through problems together. I rarely (if ever) give straight lectures to classes. That's not effective for student learning nor is it what I find interesting and exciting about teaching.

In 1110, most of the class has never written a program before. And while I do have to teach program syntax and rules, that's not what programming is about. Programming is about problem solving. It's about learning about a need that someone has, figuring out how to meet that need, designing an algorithm around the solution, and then, finally, coding the final program. The program at the end is important, but that's just one piece of the puzzle! So, my classes focus on the interaction, the problem solving, and the algorithm design that is needed to be a good problem solver and programmer. And that is not done through lecture. That is done through working together on examples in an active environment.

Curriculum Development

In the Computer Science department, I have been responsible for the creation of two brand new courses, modifications to several of our core undergrad courses, and facilitated a major change to the CS requirements with the SEAS Undergraduate Curriculum Committee.

New Courses

The first new course I created is CS 4720: Web and Mobile Systems. The course came about when I was asked what special topics course I would like to teach. I have always enjoyed web application development, but I was already teaching some of that information in my Databases course. I was then approached by a Microsoft representative about getting a class set of mobile phones for teaching a course that fall. That was when I decided to try to merge these potentially very different topics into a single course. The driving idea was that mobile devices are low-powered, always connected devices. So how do you go about doing complex things on them? You use web services, running on significantly more powerful hardware.

The course covers some basic web development, but then transitions into the creation of web services, something that was not currently in our curriculum. The web services then are incorporated into mobile applications. Students have to publish their web services for other students to use, thus forcing them to learn how to integrate unknown components. The course has been a great success, now entering its fourth year. Enrollment and interest is still extremely high and new Android tablets have been purchased for use in the course.

The second new course was created with Prof. Aaron Bloomfield and is called Service Learning Practicum. The idea behind this course is that our students are very interested in "real world projects," and we try to give them that in our courses already. However, we could fully direct this desire in a single course and focus on non-profit organizations in the area, thus filling a need in our community. This past spring, two team built projects for Appalachia Service Project in Johnson City, TN, and the Center for Non-Profit Excellence here in Charlottesville. The projects were very well received, with the Appalachia Service Project software already being used during the summer of 2012.

Course Revisions

I have modified several of our courses across the curriculum, but I believe my most significant contributions were in CS 1110, CS 2110, and CS 3240.

In CS 1110, one of the criticisms that the course has seen from other faculty in SEAS is that students could not translate their Java abilities into other languages that would apply to their disciplines. To aid this, I added a full new section to the course on transferring programming concepts to Python. In this course section, students learned how to compare two languages and why one would use one as opposed to another. Students then coded the same assignments in both Java and Python to get a feel for both how the algorithms were the same but how the syntax might be a bit different. As this is a relatively new change in the last year, I don't anticipate my colleagues in SEAS to see any differences until the 2012-2013 school year at the earliest. In this coming year, I plan on adding some Matlab to the course, which has been a request of my colleagues for some time.

In CS 2110, I was able to incorporate some of the material that I created for Web and Mobile Systems. Specifically, I introduced a project where 2110 students got to create their own mobile applications, something we thought required a bit more programming experience. It has been positively amazing to see second-semester programmers learning Android and creating their own mobile applications, and most to great success. The Android platform along with the Eclipse development environment has become an excellent way to get intermediate programmers to see what different things they can create outside of a desktop environment.

Finally, in CS 3240, I wanted to bring back projects with robots to the course, while also injecting a bit of fun into the projects. So, to do this, I incorporated Lego Mindstorm NXTs into course as the platform for the projects. However, I added some interesting twists. The main one being that the robot had to be controlled by a video game controller. Students now got to use technology that many of them already had lying around their dorm room in an actual course. They learned about Bluetooth technology and how to connect Wii Remotes and Xbox 360 controllers to their laptops. By the end of the semester, students were seeing how they could incorporate everyday technology into new systems. The student response from doing this was overwhelmingly positive.

Curriculum Changes

One of our goals since I have been on or have been chairing the Undergraduate Curriculum Committee in CS is to make the degree both rigorous but flexible. We want students to have the ability to get a minor in another subject or to really get depth in a particular area of CS. To this end, I along with the rest of the committee proposed and guided through changes in our curriculum requirements that increased the number

HSS courses and the number of unrestricted electives. This change allowed students to customize their degree a bit more, while still retaining a rigorous CS core that was in keeping with peer institutions and ABET accreditation.

I also helped shepherd through a major change in our senior thesis process by adding a capstone course to our major. Student can either opt to take the capstone sequence, which can be taught by various faculty members, or do independent research with an individual faculty member. This new addition has been a great success, with numerous projects been built each year for local non-profit organizations through the service learning capstone.

Advising and Mentoring

In my time at UVa, I feel I have taken on more than my fair share of the advising and mentoring, but I do it because I know how important it is to guide students through their years in the department. Too often, students get lost in the shuffle when an advisor goes on leave or something similar and so I want to try to help that this does not happen.

My advising assignments have been as follows:

Undergraduate Advisees

- 2008-2009 - 28 Advisees (CS, BACS, First-Year SEAS)
- 2009-2010 - 42 Advisees (CS, BACS, First-Year SEAS)
- 2010-2011 - 40 Advisees (CS, BACS, First-Year SEAS)
- 2011-2012 - 32 Advisees (CS, BACS, First-Year SEAS)
- 2012-2013 - 36 Advisees (CS, BACS, First-Year SEAS)
- 2013-2014 – 50 Advisees (CS, BACS, First-Year SEAS)
- 2014-2015 – 52 Advisees (CS, BACS, First-Year SEAS)
- 2015-2016 – 45 Advisees (CS, BACS, First-Year SEAS)

Also one of two primary contacts for all CS minors.

Senior Thesis Projects
Fall 2015 – Spring 2016

- Members of Game Design Research Group with Prof. Mark Floryan

Fall 2014 – Spring 2015

- Jennifer Lu – Gamification in Education
- Anna Greene – Gamification in Education
- (Also other members of the Game Design Research Group with Prof. Mark Floryan)

Fall 2013 – Spring 2014

- Dru Knox – Software Maintenance Lifecycles with Non-Profit Software Systems
- Travis Pennetti – Educational Video Games
- Connie Xie – Automated Quadcopters with Android Device

Fall 2012 - Spring 2013 (5)

- Michael Legore – Software Maintenance Lifecycles with Non-Profit Software Systems
- Daniel Miller – Visualizing UVa Enrollment Trends
- Amanda Ray – Efficient User Interfaces for Aiding Students with Enrollment
- Hunter Williams – Automated Quadcopters with Android Devices
- Samuel Wilson – Automated Quadcopters with Android Devices

Fall 2011 - Spring 2012 (8)

- Alex Johnson^ - Software for Non-Profit Organizations
- Matt Russell^ - Software for Non-Profit Organizations
- Erik Davis - An Evaluation of Python as an Effective CS1 Language
- Harry Bowron^ - Open-source Software for Teaching Chemistry
- Navid Hosseini* - A More Cost-Effective Unattended Ground Sensor Using Commercial Off-the-Shelf Products
- Manuel Cordovez* (SIE)- A More Cost-Effective Unattended Ground Sensor Using Commercial Off-the-Shelf Products

- Katie Hempenius* (SIE)- A More Cost-Effective Unattended Ground Sensor Using Commercial Off-the-Shelf Products
- Reed Wilson* (ECE)- A More Cost-Effective Unattended Ground Sensor Using Commercial Off-the-Shelf Products

Fall 2010 - Spring 2011 (11)

- Jared Harding - Mobile HCI for Course Selection
- Calvin Li - Voice Interface for Portable Learning
- George Washington - Voice Interface for Portable Learning
- Bennett Sorbo^ - CavDaily Advertising
- Nikhita Karki^ - Scheduling System for Student Volunteers at UVa Hospital
- Daniel Magnusson - Motion Sickness with Video Games
- Kevin Leach - Prescription Databases
- Michael Devine - Generating Music and Art using Microsoft Kinect
- Olex Ponomarenko - Generating Music and Art using Microsoft Kinect
- Derrick Brameyer - Agile Development in Student Projects (Ind. Study)
- Alan Kush - Agile Development in Student Projects (Ind. Study)

Fall 2009 - Spring 2010 (9)

- Matt Beattie - Bluetooth Identification for Vehicles
- Ryan Grigsy - Security for Emergency Announcement Systems
- Joshua Joyner - Lego Mindstorm NXT Sensor Simulation
- Seth Micalizzi - Mobile GPS Social Applications
- Charles Plucker - Mobile Virtual Reality
- Emma Rosenfeld - Teaching Time Concepts to Early Elementary Students
- John Szmuski - Bluetooth Identification for Vehicles
- Steven Trombetta - PairEval v2.0
- Jessica Vasconcellos^ - Student timecard system for Newcomb Hall

Fall 2008 - Spring 2009 (4)

- Jeffery Gaither - Web Software Source Control Management and Users
- Michael Miller - An Analysis of Static Metrics in Open-Source Software Projects
- Benjamin Plunkett - UVaCollab: Compliance with FERPA
- Joshua Sennett* - Compatibility of Partnered Students in Computer Science Education

Fall 2007 - Spring 2008 (4)

- Eric Bradbury - A New Paradigm for Tutoring at the School of Engineering and Applied Science
- Fred Dysart* - PHP Based Automated Fix for SQL Injection Attacks
- Maureen Maughn - Web-Based Kennel System in PHP and MySQL
- Danny Shih - Integrated Querying in C# using Microsoft LINQ

* denotes work published at an IEEE/ACM conference
^ denotes software created for external customer and is still in use

In addition to this, I dedicate as much time as possible to advising incoming SEAS students over the summer. I have managed numerous advising sessions over the years and will continue to do so in the future. Further, I also created materials for distribution to the first years on key resources in SEAS to help them transition.

Assessment of Teaching Effectiveness

In this section, I will provide a high-level overview of my teaching evaluations and other evidence of my effectiveness as an instructor. My complete set of evaluations can be found on my website, http://www.cs.virginia.edu/~sherriff/teaching.php.

Teaching Awards

I have been recognized multiple times for my teaching by both my students and my colleagues:

- University of Virginia All-University Teaching Award 2014
- UVa Association for Computing Machinery Computer Science Professor of the Year 2012
- Trigon Engineering Society Thomas E. Hutchinson Faculty Award Winner 2011
- Hartfield-Jefferson Scholars Teaching Prize 2010 ($12.5K, first year ever awarded)
- UVa Association for Computing Machinery Computer Science Professor of the Year 2010
- Trigon Engineering Society Thomas E. Hutchinson Faculty Award Finalist 2010

These awards include a university-wide teaching award, the largest teaching award in SEAS, twice being honored by an Engineering honor society, and twice being honored by students specifically in the CS department. I am proud of these accomplishments and very grateful to the organizations that recognized my efforts.

The Hartfield-Jefferson Teaching Award, given by the Jefferson Scholars Foundation, honors those that "exemplify the highest standards and practices of the teaching profession." Winners "demonstrate knowledge, exceptional commitment to the education of undergraduate students, and passion both for their subject and for advancing the University's curriculum." The Thomas E. Hutchinson Faculty Award "honors professors who have demonstrated a love of teaching, genuine outreach to their students, and have significantly contributed to the engineering school. The faculty award is completely chosen by engineering students."

I am also a member of the University's Academy of Teaching.

Teaching Evaluations

Overall, my teaching evaluation scores have been above average for SEAS averages in every semester and in nearly every category. My full evaluations are available on my portfolio website, but collected here are the scores from two questions that I believe accurately portray my effectiveness: "Overall, this was a worthwhile course" and "Overall, the instructor was an effective teacher."

All of my teaching evaluations are available on my website, http://www.cs.virginia.edu/~sherriff/teaching.php.

Student Comments

Below are selected student comments from each of my courses, highlighting both positive feedback and areas where improvement is still needed.

CS 1110

Positive Comments

"Professor Sherriff is an amazing professor. I looked forward to going to class every day. I don't think I've had a class at UVA that has been quite as amazing as this one."

"I genuinely enjoyed going to Professor Sherriff's lectures. He not only presented the information in an effective manner, it was entertaining and he didn't always just lecture, he let us interact. I wanted to go to lecture because I knew it wasn't going to be dry and boring and that I would actually learn something relevant to the course."

As someone who has done theater for a long while, I recognize that it takes a lot to keep several hundred people's attention for an extended period of time. I am pleased whenever students say they enjoy coming to my class.

"I had literally no programming experience prior to this course, and was concerned about its difficulty. I am doing very well in the course, largely in part because of Professor Sherriff's commitment to his students and his course. This was easily the best class I have taken at UVa."

Even though this might have come from a first-year student, it's still very nice to hear.

"Loved the course, it made me change my major to CS. Mark Sherriff is a great teacher, he is really involved in the class and easy to talk to. Even when I some trouble with submitting a homework assignment he was very understanding and pragmatic with how he handled the matter. Go Sherriff! I laughed when filling out number 19 on this form because there is no bubble for "I never got around to buying a textbook." Not saying you are a BAMF or anything... but you kind-of are....!"

"This was my favorite class that I have taken at UVA so far. The content was interesting and very useful, the lectures were engaging, and the tests were fair. I learned a ton in the class and can't wait to pursue a computer engineering major. Also, Sherriff is awesome!"

"Professor Sherriff and this class were great! Prior to taking this course, I had no interest in majoring in Computer Science. I was only taking it so that I could minor in Systems Engineering. As a result of having Professor Sherriff (and thanks to his enthusiastic teaching), I declared my BACS. Furthermore, I enjoyed this course so much, that I am taking CS 2110 with him this summer."

I love bringing in new majors.

Areas for Improvement

"Sherriff has a great personality, and truly makes class fun and engaging. That being said, I can't say I really learned anything during the lectures, but was rather introduced to concepts to which I would have to go back and learn later. I think his lecture style would be extremely effective if one already knew the material, or had learned it, and used the class as a review. I just think the lectures were so fast that it is hard to learn the material. That being said, I don't know if this is due to Sherriff or just the topic of CS. Regardless, Sherriff made the class fun, which is a heck of a lot better than many other teachers."

There are certainly times when I do get moving a bit quickly. 1110 is a difficult course in that the range of abilities is vast, and teaching to the middle often leaves some behind or makes it boring for others. I need to be more cognizant of those that might be getting left behind. I have been trying to have recitation sections for those that need more assistance, but attendance varies greatly.

"I really enjoyed this course in the beginning, but it felt like as soon as we reached loops, things started to speed up significantly and I struggled to keep up with the subject matter. I know that it is unlikely that you would be able to slow down the subject matter, because there is a lot to get through, but in the beginning everything was pretty slow and I understood fairly well, but later on in the course I felt like I was lost. One recommendation I would have is that a lot of time it seemed like we would start doing a homework and have no understanding of what we were suppose to be doing and then after the homework was due, we would practice the material that was on the previous homework. I would personally prefer if this was reversed."

In any course, timing the lectures with a pre-set homework schedule can be difficult. I do need to be a bit more aware of my timing so that things don't get done out of order. In my defense, they always do a lab on the material before the homework is due.

"While upbeat and open during lectures, one on one Mark is unprofessional and patronizing to students. His nasty attitude discouraged me from coming to him for help out of fear that I would be mocked and humiliated for asking stupid questions."

I suppose there's always someone like this, especially in a class of 484 students. I honestly don't know what happened here. Have students caught me at bad times? Certainly. Do I think I was ever unprofessional? No. I do know that my sense of humor might get me into trouble sometimes, and that's something I certainly should watch, but I'm not sure where the "Sherriff is a different person outside of lectures" comes from.

CS 2110

Positive Comments

"Possibly one of the best teaching faculty I have had, he enjoys his class clearly and every day comes with a positive attitude towards the class. He is clearly passionate about what he teaches and true to himself. He made me enjoy CS as I never thought I would specially after such a bad experience in CS 101. He made cs one of the classes I enjoyed and learned the most and even almost made me change my mind about my major. GREAT professor"

"After getting through CS 101, I thought that future CS classes would be just as dull and nearly impossible to understand, but Professor Sherriff was a fantastic professor who helped me understand the concepts of software development and more advanced Java concepts. This course changed my opinion on CS and now have an even better grasp on the Java material."

"This was a great teacher and class. It has made me feel confident about proceeding with a CS major. You can tell that Professor Sherriff is constantly trying to make the class better in any way he can and it pays off by making this one of the best classes I have had. The podcasts are extremely helpful and if they had a picture of the screen to go along with it would be perfect."

"One of my favorite professors that I have had so far. He is very animated and makes learning the material very enjoyable. I really like his analogies to make us understand topics, even though some of them were very quirky."

Areas for Improvement

"Unfortunately, I became very deterred from going to office hours after the first two times I went. It seemed as though Professor Sherriff was annoyed by some of the questions I was asking, and didn't care to look at some code/go the extra mile to really help me. Despite people always saying he was approachable and really helpful in office hours, I did not find this to be the case."

Another instance where someone apparently caught me on a bad day. I'm not really sure what I did or didn't do, but I need to make sure I'm at the top of my game in personal interactions in addition to in class. I know this is an isolated incident (as my other evaluations attest at my ability to interact with students) but I still don't want to let any one student down.

"This should be a 4 credit class"

A constant complaint from 1110 and 2110. And I agree with them.

"If possible, the podcasts would be much more useful if we could somehow see what's going on in class - even if it were just a screen shot of your computer or something."

I've considered this, but it doesn't work well with my teaching style. I don't actually use many slides anymore and just use the chalkboard. So, while the podcasts aren't the absolute best thing I could do, it seems to work well for most students.

CS 3240

Positive Comments

"Professor Sherriff consistently proves to be the best teacher I've had in all of UVA so far. Grading is fair, he answers his email quickly like a champ and the projects are grueling but awesome. CS 3240 was an outstanding class, very true to the real world of software development as I've already experienced in an internship. Death by documentation is annoying, but realistic, and the joy of seeing those darned robots finally do their thing right after an all nighter is awesome. I have plenty of reason to vent frustration here, I spent lots of late nights trying to get the stinking bluetooth to work, or trying to make that demon possessed android do my bidding, but when I step back and look at this course

as a whole, it's without a doubt the kind of course I had hoped I would take when I used to think about college in my high school days. On a side note, I recently showed a prospective CS student around UVA, and when he asked about the CS department, I was able to say without a doubt that the big three of CS (Sherriff, Bloomfield, Horton) make CS a great experience, even early in the program."

"To put how awesome Professor Sherriff is in perspective: there is approximately zero reason to come to class - no tests (besides the midterm), no quizzes, it was all the project and coming to class didn't really effect how you did on the project too much - yet EVERYONE still came. That does not happen in college."

"This was a typical Sherriff course: fun and entertaining lecture moments, some hard days (or nights) of project work and an overall great learning experience. The projects were well thought out and conveyed the ideas of the software development accurately."

Interesting to find out I had a "typical" course-type.

<u>Areas for Improvement</u>

"There are so little amount of lectures for this class. Many of classes were focused on projects. Because of this, I did not learn that much from this class that is really new and interesting. I think the professor should spend less time in talking about projects during lectures, because it wastes time. Also, I don't like the fact that professor did not have slide shows for some of the materials. Not having slide shows made some students like me harder to study for exams."

It was a project-based course, and thus I felt it did require a bit more time in class to focus on how the projects would work. I do recognize that I didn't have as many slides here, due to using a new textbook. I should provide more resources for studying in this scenario.

"All that documentation seemed like busywork."

While the documentation was somewhat the point of the course, I should probably emphasize more why it's so important.

CS 4720

<u>Positive Comments</u>

"I really enjoyed taking this course with Professor Sherriff. He was very knowledgeable and enthusiastic about the material he presented to us, which in turn made it very easy to learn and take a lot out of this class. His engaging teaching style made attending lecture an incredibly enjoyable experience and the course helped teach all about web and mobile frameworks and architectures. I highly recommend this course to anyone interested in the Internet or mobile phones. Professor Sherriff is easily one of the best professors I have had at the University."

"Professor Sherriff did a great job at teaching this course. His lectures were engaging and easy to follow. The project was fun and helped us learn a lot about Android development. I feel that I learned a lot about development on both web and mobile platforms and am better prepared for future interviews/internships."

"This was my first time taking a course with Professor Sherriff, and it was without a doubt the best CS course I have taken thus far. The information you will learn in this class is incredibly relevant in today's technology and is extremely useful for real world applications. From learning the intricacies of web development to building your own Android App, this class pretty much covers it all. It is certainly a fast paced course, and you will have to learn a bit on your own, but Professor Sherriff's enthusiastic lectures and passion for teaching definitely makes this class worth it!"

"Since course-specific questions were covered previously, I just want to make a few comments about Professor Sherriff. This is an immensely popular class, and for good reason; the subject material is engaging and fulfilling, and it is exceptionally well-taught. Professor Sherriff is everything one expects

from teaching faculty at a large university, and more; he makes a significant effort to go above-and-beyond his duties as a purveyor of material. Sherriff includes interactive activities in the classroom, and makes himself constantly available for extra help. I found that I was always able to receive a prompt reply to emails, and my questions were answered thoroughly and to my complete satisfaction. The classroom teaching style is somewhat informal, but the appeal of this, I suppose, is somewhat subjective. While there were sometimes tangents which seemed irrelevant in class, they usually ended up adding to the overall material; if nothing else, they certainly made Professor Sherriff more personable and approachable. So, overall, Professor Sherriff is one of the most effective teachers under whom I have ever had the pleasure to be a pupil, and I consider myself extremely fortunate to have experienced this course."

<u>Areas for Improvement</u>

"Please please please please please grade things faster! Really. It was the one thing that kinda made me a bit nuts."

"As I said before, would have liked to get feedback on the project earlier so that we could have made changes on what you didn't like"

This course had 83 students in it and taught me that this course couldn't not go this large again. It did affect the quality of the instruction, mainly the response time to assignments.

"I loathe RoR, with a passion!!!!"

RoR stands for Ruby on Rails and was a platform choice I made for the class. I probably won't choose that again as it created a number of headaches for me and the students, and did lead me to be short with a couple of students, unfortunately.

CS 4750

<u>Positive Comments</u>

"Sherriff makes coming to class at 9:30 in the morning exciting and worthwhile."

"One of the most practical courses I've taken at UVa. I'm sure the material covered in this course will prove to be highly useful in my future career."

"Mark Sherriff is one of the best professors I have ever had. He works hard to incorporate technology in the classroom and engage his students. It has been an honor to be his student."

"Prof. Sherriff is one of my favorite teachers at UVA, and easily one of the top teachers in the CS department. His lectures are lively and entertaining, even at 9:30 in the morning. He is clearly knowledgeable of the subject material and frequently makes connections to real-world applications, which is very helpful. Of all the other classes I've taken in the department, I find that I pay the most attention to his. His passion for the subject is infectious, and I hope to take more classes of his in the future."

<u>Areas for Improvement</u>

"You can lead a horse to water but you can't make it drink; I would suggest having more "checkpoint" like assignments on the project to discourage people from putting it off until the last minute in the future. (Show me the stardock print out of your schema, show me a screenshot of a page to login/access data/do a report, etc). With the understanding that things will probably still change after that checkpoint but just so that you know students HAVE created their tables at least, and HAVE done a little bit of web coding."

Again, in a large class, I found it hard to make people accountable for certain things without the assistance I needed. I needed to facilitate this better.

"Sherriff is a great teacher, and he did a solid job teaching the material. The one misstep, as I see it, was the SQL coverage--particularly the questions on the test. There was one question in particular which I asked Prof. Sherriff about; he told me he couldn't help me with it and sent me to the TA. I asked the TA, and even he was hardly able to do it. The question was clearly unreasonable to ask on a test; I'm certain Sherriff himself couldn't have done it off-the-cuff in an exam situation. The course was overall really solid. Other than the occasional misstep, I was glad to take it. I did feel it could have been a bit more intensive, particularly as regards security matters, but even so I learned a lot."

I might have stretched them a bit on a few questions. And this student is right – I might not have been able to do the question "off-the-cuff," but I was more interested in seeing their thought process.

"I'm not a fan of project-based learning where most of the effort goes into things unrelated to the course material. "But those things are still important!" you might object. So is physical fitness, but we're not going running and doing push-ups for homework... The scope of the class is databases, not web design, not software engineering. We should be learning about databases! The same thing happened in the CS3240 project; so much effort went into the specifics of the project, such as snapping Lego blocks together, that ideas like requirements analysis and rigorous testing -- the actual course material -- became a secondary hassle. In CS4750, reasoning about databases was only a tiny part of the project... that means something is wrong! Even the things we did learn about databases were often superficial and vocational. We should be focusing on the internals of the DBMS before worrying about specifics of SQL. Maybe people in industry are asking for graduates that know SQL, but, as Dijkstra said "it is the task of the first-class University to tell industry what it does not want to hear" (EWD 1165). Google can teach me about tools. University should teach me the fundamentals."

I'm not really sure how to respond to this one. I feel the material was definitely about database. True, the project required some software engineering concepts, but so does any software construction exercise. I do come from a software engineering background, so it does make it tougher to separate some of those lessons from other courses.

Efforts to Improve Teaching

I do several things to try to improve my teaching each semester. These fall into a few discrete categories: iterative course feedback, external review, and professional involvement.

Iterative Course Feedback

My main avenue for providing feedback to myself is through student comments and feedback. I always leave anonymous feedback turned on, regardless of what I might get back from it. Further, I add numerous extra questions to my course evaluations each semester to provide more fine-grained information about particular parts of my courses. Students are always welcome to send me feedback in any form.

Another technique I use to provide more feedback to myself is through my course podcasts. While I certainly do the podcasts for my students' benefit, they are also invaluable to me as well as I go back and listen to what I covered on a given day. It's also very helpful to be able to go back to previous instances of a course to hear exactly how I presented material in other times.

External Review

Every semester, my faculty mentor, Prof. Tom Horton, comes and observes my teaching in at least one of my courses. His guidance and feedback over the past five years has been invaluable. His years of experience and calm demeanor have made him an absolutely perfect mentor for me and I could not underestimate his contributions in guiding me in my career to this point.

Other faculty in CS have also had the opportunity to view my teaching. Profs. Aaron Bloomfield, Jim Cohoon, and Marty Humphrey have each seen me teach under varying circumstances. Their feedback has also been quite excellent, and I thank them for their assistance.

Professional Involvement

While at UVa, I have availed myself of the resources both at UVa and in the greater CS education community. The seminars and programs offered by the Teaching Resource Center helped me get my footing here at UVa and were quite useful in the first years of my career. Outside of UVa, I routinely attend and present at educations conferences in both computer science and engineering as a whole. These experiences have provided valuable external input on my teaching methodologies and have given me numerous ideas about how I could improve on my own teaching.

Links Between Teaching and Research

My overall goals with my CS education research are establish the best ways to do large projects in large software development courses. Thus, my research here and my research proposals are often about creating teams, teaching teamwork methods, and project management in courses. I have submitted two NSF proposals recently as the PI specifically related to teaching teamwork in engineering courses.

Every year, I attend the SIGCSE (Special Interest Group on Computer Science Education) Symposium, the main conference in CS education. At this conference, I organize and run the NSF Showcase with Prof. Aaron Bloomfield. This gives me an opportunity to work with education researchers from around the country and with representatives of NSF. These interactions have been quite valuable in shaping some of my teaching. Further, I attend the ASEE/IEEE FIE (Frontiers in Education) conference when I can, exposing myself to a broader range of educators.

While I do enjoy attending these conferences, I do also go to present my own work. I have written several papers on CS education topics that have been published in various venues:

- Al-Zubidy, A., Carver, J., Heckman, S., Sherriff, M. "A (Updated) Review of Empiricism at the SIGCSE Technical Symposium." The 47th ACM Technical Symposium on Computer Science Education, Memphis, TN, Mar 3-6, 2016.
- Bloomfield, A., Sherriff, M., and Williams, K. "A Service Learning Capstone Practicum." The 45th ACM Technical Symposium on Computer Science Education, Atlanta, GA, Mar 5-8, 2014.
- Layer, R., Sherriff, M., and Tychonievich, L. "Inform, Experience, Implement - Teaching an Intensive High School Summer Course." The 42nd Annual Frontiers in Education (FIE) Conference, Seattle, WA, Oct 3-6, 2012.
- Sherriff, M. "Teaching Web Services and Service-Oriented Architecture using Mobile Platforms." The 40th Annual Frontiers in Education (FIE) Conference, Washington, DC, Oct 27-30, 2010.
- Krogius, O., Horton, T., and Sherriff, M. "Role of Larger Software Artifacts in Introductory Computer Science Courses." The 40th Annual Frontiers in Education (FIE) Conference, Washington, DC, Oct 27-30, 2010.
- Lew, M., Horton, T., and Sherriff, M. "Using LEGO MINDSTORMS NXT and LEJOS in an Advanced Software Engineering Course." The 23rd Annual IEEE-CS Conference on Software Engineering Education and Training, Pittsburg, PA, Mar 9-12, 2010.
- Sennett, J. and Sherriff, M. "Compatibility of Partnered Students in Computer Science Education." The 41st ACM Technical Symposium on Computer Science Education, Milwaukee, WI, Mar 10-13, 2010.
- Burg, J., and Sherriff, M. "Unix Tutorials to Move Students from PC/Windows to Unix." ED-MEDIA 2002 Conference, June 30, 2002.

These papers are available on my website. Each of these papers highlights a different teaching technique, methodology, or course that I implemented at UVa, showing the recognition by my peers of my work in CS education.

Further, my Encryption Chase assignment was recognized at the 2010 SIGCSE Symposium as being one of the "Nifty Assignments" in computing that year.

Senior Thesis Projects

While I have directed several senior thesis projects, some were specifically in CS education, specifically:

- Erik Davis (2012)- An Evaluation of Python as an Effective CS1 Language
- Derrick Brameyer and Alan Kush (2011) - Agile Development in Student Projects (Ind. Study)
- Joshua Joyner (2010) - Lego Mindstorm NXT Sensor Simulation
- Emma Rosenfeld (2010)- Teaching Time Concepts to Early Elementary Students
- Steven Trombetta (2010) - PairEval v2.0
- Joshua Sennett (2009) - Compatibility of Partnered Students in Computer Science Education

I'm very proud of the work these students did and I hope to advise more projects like this in the future.

Dissemination of Teaching Materials

There are two main areas in which I have been proactive in sharing my teaching materials – pair programming and the encryption chase.

My work and advocating for pair programming (an extension of my work with my PhD advisor at NC State University, Dr. Laurie Williams) has had an impact both at UVa and at other schools. I have shared my knowledge of team creation and management with several other SEAS faculty at UVa and have given several presentations about it to over 100 high school teachers. They have also contacted me to let me know how their efforts have been going. I'm proud to say that my efforts in spreading pair programming as a mechanism for helping lower-level programmers has been successful.

I have shared my encryption chase materials with both organizations inside UVa, with other schools, and with industry. This past summer, I used the chase materials with two one-week middle school programming camps, sponsored by Google. After seeing what I did with this program, representatives at Google asked to use my materials for their other camps over the summer.

Concluding Remarks

Ever since I entered graduate school in 2002, I have focused my attention and career goals on being an excellent teacher. At NC State, I TAed several large courses, earning a TA award for my efforts. I became the instructor of record for a summer databases course, that I was allowed to teach for four consecutive years. I also took coursework specifically on teaching college courses. My advisor, well aware of my desire to teach, allowed me to focus my time appropriately between research and teaching, which allowed me to build up my portfolio and get a job in teaching at a great school like UVa.

When I arrived at UVa, I was definitely new to the profession, but I came in eager to learn and ready to get into the classroom. I certainly had some problems along the way, as one course in particular didn't fit my style particularly well. I learned, though, where my strengths were, and with the help of the department, I was able to focus on large, software system courses that I know well. I was also given the freedom to innovate on the curriculum and bring in new ideas and topics when I saw there was a gap in our courses.

I believe that I am a good teacher. I believe that my classroom is an environment that students actually want to come to in order to learn and that they do learn while they are there. I also believe that my classroom is an entertaining environment, because learning can absolutely be fun. I am ever grateful to the students and colleagues who have given me feedback on my teaching, and I continue to hope that I will improve as a professor as my career moves forward.

Supplemental Materials

Available at: http://www.cs.virginia.edu/~sherriff/

On this page, you will find the following:

- Links to all my papers
- More information about my dissemination of materials, including examples
- Links to my course websites
- All of my course evaluations
- Links to the podcast feed for many of my courses (some are now built into the course websites)
- An unsolicited letter from a secret society at UVa about my contributions to the University

Example course materials:

CS 1110 (Introduction to Programming) homework assignments:
https://cs1110.cs.virginia.edu/f15/oldpotd.html

CS 1110 (Introduction to Programming) lab assignments:
https://cs1110.cs.virginia.edu/f15/oldlabs.html

CS 1110 (Introduction to Programming) lecture notes and audio recordings of lectures:
https://cs1110.cs.virginia.edu/f15/lecture/

CS 4720 (Web and Mobile Systems) Android project:
https://cs4720.cs.virginia.edu/f15/android.html