CS 2110-100 **Software Development Methods** - Spring 2010

ENGR (10285)

INSTRUCTORS: **Sherriff, Mark (mss2x)**

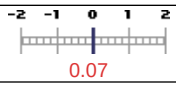
Respondents: 83 / Enrollment: 108

Summary: CS 2110-100 Software Development Methods - Spring 2010 (10285)

Overall Course Rating

CS-2110-100 Mean 4.05
CS-2110-100 Std Dev 0.92
CS-2110-100 Response Count 408

Difference from Category Mean, Expressed in Category Standard Deviations

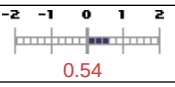


SEAS, 2000-level courses Mean 3.99
SEAS, 2000-level courses Std Dev 0.97
SEAS, 2000-level courses Response Count 12684

Overall Instructor Rating

INSTRUCTOR: Sherriff, Mark
Mean 4.63
Std Dev 0.55
Response Count 577

Difference from Category Mean, Expressed in Category Standard Deviations



SEAS, 2000-level courses Mean 4.13
SEAS, 2000-level courses Std Dev 0.92
SEAS, 2000-level courses Response Count 18059

~ QUESTIONS AND DETAILS ~

~ ANSWER MATRICES ~

1. How accurate is this statement for you: After taking this class, I am more likely to major or minor in CS.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	3.75	1.20	28 (33.73%)	26 (31.33%)	12 (14.46%)	14 (16.87%)	3 (3.61%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	3.75	1.20	28 (33.73%)	26 (31.33%)	12 (14.46%)	14 (16.87%)	3 (3.61%)

2. How accurate is this statement for you: After taking this class, I have a better appreciation for Computer Science.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.46	0.63	43 (51.81%)	36 (43.37%)	3 (3.61%)	1 (1.20%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.46	0.63	43 (51.81%)	36 (43.37%)	3 (3.61%)	1 (1.20%)	0 (0.00%)

3. How accurate is this statement for you: After taking this class, I personally have a better understanding of fundamental concepts in Computer Science.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.46	0.65	44 (53.66%)	33 (40.24%)	4 (4.88%)	1 (1.22%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.46	0.65	44 (53.66%)	33 (40.24%)	4 (4.88%)	1 (1.22%)	0 (0.00%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

4. How accurate is this statement for you: Pair Programming helped me learn the material better.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.65	1.24	26 (32.10%)	22 (27.16%)	17 (20.99%)	11 (13.58%)	5 (6.17%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.65	1.24	26 (32.10%)	22 (27.16%)	17 (20.99%)	11 (13.58%)	5 (6.17%)

5. How accurate is this statement for you: The project was of acceptable length.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.16	0.69	25 (30.12%)	48 (57.83%)	8 (9.64%)	2 (2.41%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.16	0.69	25 (30.12%)	48 (57.83%)	8 (9.64%)	2 (2.41%)	0 (0.00%)

6. How accurate is this statement for you: The project was of acceptable difficulty.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.09	0.76	23 (28.05%)	47 (57.32%)	8 (9.76%)	4 (4.88%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.09	0.76	23 (28.05%)	47 (57.32%)	8 (9.76%)	4 (4.88%)	0 (0.00%)

7. How accurate is this statement for you: The project helped me better understand the phases and intricacies of software development.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.16	0.76	29 (35.37%)	39 (47.56%)	12 (14.63%)	2 (2.44%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.16	0.76	29 (35.37%)	39 (47.56%)	12 (14.63%)	2 (2.44%)	0 (0.00%)

8. Which topic/lecture in this course was your favorite and why?

~

Question Type: Short Answer

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Individual Answers
76	*See below for Individual Results*

I enjoyed the lecture on Encryption and the race. It was challenging and fun.

I liked learing about how to breech the security of websites

Encryption -> bonus points

Networking was cool because we all use the internet without understanding how it works.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I personally found the GUI and Security lectures very interesting, because they really expanded my general knowledge, and showed interesting parts of programming that we had not really discussed. Android project Although it was a shorter topic, I really enjoyed the encryption and cipher information. My favorite major topic had to be algorithms because it is very helpful to know what can make the program run slower and faster. Sorting, because it was interesting how there were better and better ways to use it. Use-Case modeling because the concept was pretty simple and easy to understand. I honestly enjoyed everything. I really liked the lecture on hacking and internet security. This was really fun and it was neat to see how what we had been learning about all semester was related to more than just typing code in eclipse. I really enjoyed learning GUI/jigloo code. I like being able to see my programming efforts presented in an elaborate window with menus, buttons, and text fields. I found networking to be the most enjoyable topic - it's an area I have always been interested in. The lecture on hacking was my favorite; it was something interesting to consider as far as Internet security went. The blue bucket lecture. I liked learning all of the libraries with the Android event driven programming, because we get to work with actual stuff for users I was really impressed about the 5 faces of development, which could be used in all other fields also. My favorite part was working on the project and learning about Droid programming. Why, you ask? My opinion may be slightly biased due to the fact that I am absolutely obsessed with Google and all its affiliated products! Hacking encryption, because we got to do a scavenger hunt. Hacking was pretty fun because, well, hacking is just cool! SQL Injection my favorite was the project on droid phone...it was AWESOME! The part about use-case diagrams would good because I finally knew what was going on in my Systems class with regards to use-case diagrams. the hacking session was very interesting Internet security/ encryption b/c the chase was fun (although extremely long and excruciating annoying due to the May heat) The Chase was fun. The project, event driven programming: interesting to see how programming can work in a real life situation SQL injections...because I always wondered if you can inject code into input fields. Trees, because I love trees. So simple and easy and their solutions are usually so elegant, while mildly challenging to figure out. Tree. It is very useful The project because we got to program as we liked to meet requirements. Design and Requirements The encryption one was great. Mostly for the puzzle solving aspect of it. The hacking class was cool. I would have liked to spend a little more time on that (even though it's just a bonus subject and not really required). iterators because it was different and seemed very practical for when programming also i liked the examples given in class with the blue boxes :)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	hacks, because it was interesting
	internet security it's fun and misterious!
	My favorite topic was learning about event driven programming because I have had little exposure to that topic before.
	I found networking to be the most interesting topic. It is the topic most closely related to what I want to do and understanding how networking interacts with Java is very helpful.
	Doing application for android phones because we got to learn how real software developers work against a deadline and requirements.
	SQL Injection because I am interested in cyber attacks and internet security
	The hacking lecture was a bunch of fun
	I thought the Android topic was too much fun.
	Maintainence- it was very easy to understand the different types of maintainence through the lecture.
	Encryption and hacking because they were the most enjoyable topics and because I could brag to my friends about the material I was learning.
	Threading because I found it most interesting
	Class diagrams because gives a big-picture approach and doesn't take coding.
	The different phases of CS because I like to lear about systematic approaches.
	Internet Security bc identity theft is so prevalent
	Going over the individual phases of software development.
	I enjoyed all of them
	group project: allowed us to work on our own w/ appropriate help
	Encryption! I find it really interesting, and I would love to learn more about how to apply CS concepts to the different techniques of encrytion
	I enjoyed threading because it was something that I haven't done much before
	I enjoyed exploring and learning about development for the Android OS.
	The hacking course seems very interesting
	I liked learning about the different types of testing. This was new material to me and I feel like the professor presented an engaging lecture on the topic.
	encription, got to run around outside and got extra credit
	Internet Security/Hacking
	I really enjoyed learning about binaray search trees because they are a challenge to wrap your mind around at first.
	My favorite topic was inheritance and polymorphism because I find it interesting how they can be used to model systems and for code reuse.
	The android project, it was fun to see that we could produce something like that
	I really liked the project because I enjoyed learning how to use my cs skills in a practical way.
	The most interesting portion of the course was definitely the end project, learning of the testing process, and how CS practices are implemented in the corporate world.
	I really liked everything but I think internet safety is extremely interesting
	internet security and encryption because they are the most fun
	The recursive data structures section was very interesting. The different ways to organize and maintain data intrigued me.
	Phases of Software Development or Internet Security
	Network security. It was just really interesting to see how SQL and XSS worked.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Event driven programming has always seemed mysterious and scary and now I understand (at least at a basic level) how it works. word! the last few lectures were my favorite because they covered interesting topics about the internet, security, and how computers work. Hacking, it was a fun activity and I have always wondered how some of it is done. Hacking-Internet Security The Android project.
9. Which topic/lecture in this class do you think you will find the most useful in the future? ~ Question Type: Short Answer ~ *contributed by Sherriff, Mark (mss2x)*	Results for CS-2110-100, Sherriff, Mark Total: 72 Individual Answers: *See below for Individual Results*
	Internet Security The Agile Development lecture was very interesting. I have been able to apply some of the stuff they told us to other aspects of my life (ie, fraternity exec meetings, etc) Probably the Project; it was a challenge learning the specifics of the droid code and we had to learn it very quickly and on our own for the most part--as the TAs were essentially learning with us. Overall the whole class clarified my knowledge of Software development and was useful in every sense. Probably the phases of development. Just knowing how a company goes about designing things is a huge advantage As I am not going into CS in the future, none of them will probably be useful. I enjoyed them all though. Encryption Programing for a DROID. This was a great (though difficult)experience where I will be able take away not only the programing itself, but the process of having to search the internet to learn how to code it myself, which as I hear is the way you would always have to in the "real world" anyways. The levels of development and the testing process. Threading JCF/ADT, threads, 5 phases of software design threads,internet security, the project Class Diagrams, Stages of software development, UML, etc. Basically anything that could be used to my advantage in a job interview. Testing because it will save or cost me time in the future Agile development After completing the project i now feel capable of actually programming something useful security Android Project I think complexities, 5 faces of development- approach to a project, and ADTs would get more important in future. Trees. Phases of Software Development (business aspects) 5 Phases of Development All of them Working in groups

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I think that my gained understanding of sets and maps will be used in the future.
	The methods for communicating ideas (class diagrams, uml, etc) will probably be most useful, but I'm in engineering science so I don't know how much I'll ever really use.
	I think that class diagrams will be most useful in the future when I have to design complex systems.
	Many of the lectures I feel will be useful in the future.
	networking
	networking
	I think ADTs and how to implement them and manipulate them will be very helpful to me in the future.
	The topic of the project involving the Android will have the most useful purpose in the future because it has helped me gain more experience to what working with other types of technology feels like.
	Learning about multithreading helped a lot, because when I had learned Java earlier, this was a topic I had trouble learning.
	the use of subversion for group programming projects
	The IM project.
	The lecture on Maintenence
	The overall learning of the software development cycle.
	probably the 5 phases of development, "keep it in your bag of tricks"
	Learning about the different Abstract Data Types was extremely useful to add to my bag of tricks - ADTs are a fundamental part of programming and one needs to have proficiency in this area.
	the project section of the course was very applicable to future situations especially with programming software for companies
	The 5 stages of development and Scrum because they are topics applicable to non-CS majors as well as CS majors.
	Networking
	The project.
	probably the design elements/actually working through them
	networking? i don't know...
	Android, currently working on writing a tank game that I plan on adapting to the android.
	The software design process, as it describes how programming is actually done in the real world.
	Five phases of development- it shows the overall understanding of any computer science- related business, and gives a good picture of how things work at the macro level.
	I liked the big picture lectures like the stages of development lectures at the beginning of the year. As a Systems major, I didn't care as much for the programming lectures but I liked the lectures that are more borad and project management based. On the same note, I really liked Mr. Butler's guest lecture because it was very tangible for non computer science people.
	Agile development and the phases of software development.
	Recursion and data structures.
	Working together in teams, and the team-building skills are useful.
	Internet security
	learning the android sdk
	Threading.
	Threading.
	none, I'm pre-med
	Networking/security. very useful to know how exactly does networking work.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I really liked the guest lecture on Human Computer Interaction. What he said about usability applies to all programming and the design process in general. The ones on the big ideas in programming, ie abstraction, were really helpful in seeing programming in a different way. But I can honestly say that everything from the class was useful. 5 phases of software development I feel like threads and networking are topics that will probably be very useful to have seen in the future. I think that UML and the related lectures will be the most useful in the future. The majority of the material covered I could learn myself outside of the class, but the UML is not something directly relating to my interests. Being forced to learn the material taught me things I would not otherwise pursue. stages of software development Testing will probably benefit me the most in this class, because I have had very little exposure to it in the past, and I'm sure it will be something that comes up again and again. I think just learning so much about the fundamentals of software design will help me the most Learning about maintenance because that is where most of the budget is spent so reducing maintenance would lower costs for firms. Algorithms Hacking-Internet Security The Android project. I thought the information on networking was interesting. All of the information on design and testing will undoubtedly be useful. The lecture from Innova was useful in seeing what CS majors actually do in the real world. Class diagrams and program initial design.
10. What lecture/topic(s) in this class "did not work" or were not seen as useful in the long run? ~ Question Type: Short Answer ~ *contributed by Sherriff, Mark (mss2x)*	Results for CS-2110-100, Sherriff, Mark Total: 64 Individual Answers: *See below for Individual Results*
	I highly doubt the hacking stuff will be at all useful. Interesting though. There didn't seem to be a topic that "did not work", they all are part of Java and other lectures had real world application. I didn't love the project because I didn't know any XML and was frustrated by it a lot n/a n/a guest lecturers. It would have been helpful if we could have spent an additional class on threading - for whatever reason I found threading to be confusing. I wish we spent more time on the networking lectures. I had trouble taking solid notes for that section based off of the lecture. The topic of trees was never really connected to real world applications. I understand what trees do, however, I would have liked to know when you are supposed to use them. Java. I want to learn C++. The networking topic was so short that I don't think I really got anything from it. It served as a really basic introduction to the terms, but that was about it. The testing homework was dull and unenlightening. I understand the importance of testing software, but there has to be a more interesting way to learn it.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The only thing that comes to mind could be the recursion section, but that is only because I had seen it before in at least two classes. I'm sure if it was my first exposure to it, it would have been more helpful.
	Trees
	Not applicable. All lectures were amazing.
	The GUI programming unit was a little weak. We probably should have at least had a homework on it.
	I'm not sure
	none
	none
	none
	none
	All seem useful.
	everything worked well
	most all were applicable to the final goals
	None come to mind.
	UML diagrams were not helpful for me.
	Almost everyday had a new topic/lecture that added to understanding homework and more.
	I do not think that I will use knowledge of the software industry much after college, as I am not planning on majoring in CS.
	I can't think of any.
	i dont know
	No lectures come to mind, but on the topic of paired programing; On several ocassions my partners (who were in the college and thus had tons of free time) would come to our group meetings with huge sections of the code already done. Although this was nice, I wasn't able to learn hardly anything from those homework assignments. I think paired programing is extremely useful and more efficient but for the sake of the course I would recommend alternating back and forth between paired and unpaired (which you can ask classmates questions) assignments.
	networking
	nothing in particular
	Algorithm complexity did not seem very useful.
	Maybe World of Warcraft on those few occasions the topic came up. I might think differently if I was an avid gamer.
	I felt like most of the topics where well covered and meaningful. However, I felt like trees where the most difficult programming wise, and was kind of hard to grasp.
	I think all of them are helpful
	The end of the class (Networking/Security) was a bit confusing. I thought you went too fast through the topics and assumed we knew more than we did.
	None
	None
	We learnt a lot of topics. I don't think any of them is unimportant or didn't work.
	JCF
	Trees could have been covered a little more in depth.
	I did not see real applications for the HaspMap code and homework.
	Trees were lost on me...
	N/A

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

N/A

N/A

"Hacking" the server, while interesting, didn't seem to fit in with the rest of the course.

Nothing comes to mind.

I think that the only thing that did not work was the lectures on testing just because I felt we needed to see more examples of it.

I thought everything worked. Threading was a little short, I didn't really understand exactly how notify and notify all worked or how the scheduler knew what to notify. I guess a little more of the mechanics would have been helpful.

nothing really

threads were extremely confusing

I didnt like the guest lectures. They were usually bad.

all worked

the testing using JUnit tests.

After having taken this course, I still feel like I do not understand Trees/Recursion very well.

None.

HUI (Human User Interface)

N/a

the topics talked about during the last week before the final...because they were interesting but only 1 -2 lecture(s) for each topic was not enough to understand them well

There could have been more extensive coverage (i.e. more homework) on trees, because while I understood how trees work when we go through the code in class, I couldn't understand how to code them myself.

Algorithms

11. How accurate is this statement for you if you used the podcasts from this class: Podcasts were useful to catch up on material that I missed due to absences.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark								
Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.22	0.81	23 (28.40%)	16 (19.75%)	12 (14.81%)	0 (0.00%)	0 (0.00%)	30 (37.04%)

Results for SEAS, 2000-level courses								
Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.22	0.81	23 (28.40%)	16 (19.75%)	12 (14.81%)	0 (0.00%)	0 (0.00%)	30 (37.04%)

12. How accurate is this statement for you if you used the podcasts from this class: The podcasts were useful to review material that I was unclear on.

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark								
Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
82	3.77	1.00	14 (17.07%)	17 (20.73%)	18 (21.95%)	1 (1.22%)	2 (2.44%)	30 (36.59%)

Results for SEAS, 2000-level courses								
Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
82	3.77	1.00	14 (17.07%)	17 (20.73%)	18 (21.95%)	1 (1.22%)	2 (2.44%)	30 (36.59%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

13. How often did you listen to the podcast for a lecture?

~

Question Type: Multiple Choice

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Every lecture (NA)	Nearly every lecture (NA)	Whenever I needed to review a topic (NA)	Only when I missed a class (NA)	Randomly just to see what it was like (NA)	Never (NA)
82	0 (0.00%)	1 (1.22%)	17 (20.73%)	23 (28.05%)	11 (13.41%)	30 (36.59%)

Results for SEAS, 2000-level courses

Total	Every lecture (NA)	Nearly every lecture (NA)	Whenever I needed to review a topic (NA)	Only when I missed a class (NA)	Randomly just to see what it was like (NA)	Never (NA)
82	0 (0.00%)	1 (1.22%)	17 (20.73%)	23 (28.05%)	11 (13.41%)	30 (36.59%)

14. Do you have any suggestions/comments that we should take into account for future projects for this course?

~

Question Type: Short Answer

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	Individual Answers
61	*See below for Individual Results*

no.

n/a

It seems to me that there should be one project, and I picked the IM project because it could theoretically still be used after it was over

Start early and do not procrastinate

It was hard to learn the language necessary to code the android.

Teach a little more android before giving a project that uses it. And teach listeners better.

Teach the class to use subversion, because it was a very helpful tool for our group, but did require a few steps to get it working properly, and my group experienced a few problems at first.

I think only having one type of final project (as opposed to IM vs droid) would be good. The teacher could direct the lectures more at one particular idea.

I liked the project, but I felt that the Android's learning curve was really steep and we should have been allotted more time accordingly.

The Android project was really neat and fun, and now I know never to buy that phone because its battery life sucks.

Don't do two separate projects - everyone should do the same one.

Nope.

Make sure the TAs know how the Android phone works! Also, make sure they actually have office hours the week before the project is due.

Continue the Droid project!

no

Give more instructions in class on Droid GUI/APIs

more test code needed for the android project

I was on an android team. I loved the topic, but due to technical problems, waiting on the phones to arrive, example code to be posted, and the TA's all trying to learn it too, my group had to four weeks worth of work in a week and a half. There was nothing anyone could do in our class, but Next time, make the basic information about programing for android available sooner. :)

A more in depth coverage of GUIs at some point before the project begins would have been helpful.

Some people in a group just aren't going to work because they feel they can ride off their partner's abilities, which also forces more work onto the other partner.

make everyone do android.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	To cover more about the creation of GUIs, combining both Java and xml.
	Don't do two projects at once again.
	For the podcast- it'll be better to record the lectures (slides included), instead of just using the podcast, like how prof horton does. It'll be nice to know what the professor is talking about For the android project- I felt that more android concepts could be taught. we sort of didn't know how to start/copying code without really understanding it
	No.
	No.
	The project was one of my favorite parts of the class and I feel as though it is as close to perfect as you guys can make it. Sometimes it seems like groups are not as equal experience wise but that is almost uncontrollable. Also when beginning the project it is somewhat overwhelming but the resources and TAs offer great guidance so that it was not as rough as it could have been.
	KEEP USING ANDROID!!!!!!111!!11!1!!1ONE!!1
	Definitely have just one project going on at a time.
	I feel like a better resource to understand how the phone OS works with intents and switching between screens would of been useful. Its not exceptionaly difficult but we spent alot of time doing nothing because we diddn't understand the OS.
	i don't think you should do dual projects because it seemed like the TAs were spread thin. seemed like there was more focus on one project than the other.
	Android is awesome, and developing an application was one of the best experiences I've had this year. I'm planning on using what I have learned to develop an Android/Iphone app this summer (boo objective C). I strongly recommend leaving in the option to develop an app.
	I would still suggest droid project is the best!
	If you are going to do the android application project again, please spend some time in class going over the basics of android programming.
	More time for Android to teach ourselves what is going on.
	I would say that you should maybe explain the android software a little bit more. I understand that it was the first time doing it though, so I totally understood the lack of background on how to write the code.
	In my opinion, we should have had MUCH more emphasis on good coding conventions - comments, naming, good class structure.
	Don't do two projects
	No
	nope
	nope
	N/A
	N/A
	Possibly more projects where you can choose your partner would be nice.
	instead of just RAFs make us show our code each week to the TAs to be sure stuff is getting done.
	Talk about XML before the project starts so that everyone has some knowledge base
	There is a little application that comes with the android sdk that lets you override coordinates from the computer maybe that could be a better way of implementing a demonstration?
	break up the group project more
	Having two different projects does split up the resources, so that would not be the best thing to do in the future. Other than that, projects were good with lots of resources to work with as well.
	Nothing comes to mind.
	The project was perfect as is, and I loved learning about Android.
	Please handle project team problems more effectively. It really put unnecessary pressure on other team member and is very un-fair.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

Consider the fact of having to learn a whole new framework of code; the android structure seemed really different than the Java framework as a whole; with both the students and the TAs having to learn it at the same time we were often frustrated by problems.

Going into Android programming the way we did (the blind leading the blind style) was kind of intimidating, but I'm sure that would not be an issue if the Androids were used in the future, just because we all have more experience with them now.

I think that the whole class should work on the same project, instead of splitting it to two groups.

I think when you have two different project letting people pick between can hurt. Because one project might have all strong programmers and leave the other project with people who are not that great and have them weak and be hide.

Please tone down the difficulty level, or spend more time preparing students for the project during class time or during lab, because most of the time, I felt quite lost.

I know it was the first time doing the phones, but a lot of the time, I felt like we were expected to know/figure out things that hadn't even been discussed in class.

Project topics are awesome, but I think there should be a better way of assigning partners.

I felt like there was little help from the teaching staff for the project. Students were expected to self teach on completely new concepts. While this was somewhat beneficial, it was time consuming and stressful, especially for those who had multiple other time consuming assignments to worry about. Having one full class completely dedicated to presenting examples on how to compile the project would have been helpful.

The project, at least my group, was dominated by the CS majors. I wanted to help more, but they new so much more than I did that it was very hard for me to follow them. I don't really know how you could change that.

15. During the project, how many hours per week did you dedicate specifically to project work?

~

Question Type: Multiple Choice

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-100, Sherriff, Mark

Total	0-2 (NA)	3-5 (NA)	6-8 (NA)	9-12 (NA)	13-16 (NA)	17 or more (NA)
83	7 (8.43%)	34 (40.96%)	21 (25.30%)	12 (14.46%)	7 (8.43%)	2 (2.41%)

Results for SEAS, 2000-level courses

Total	0-2 (NA)	3-5 (NA)	6-8 (NA)	9-12 (NA)	13-16 (NA)	17 or more (NA)
83	7 (8.43%)	34 (40.96%)	21 (25.30%)	12 (14.46%)	7 (8.43%)	2 (2.41%)

16. The subject matter was challenging.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
83	3.96	0.83	16 (19.28%)	56 (67.47%)	6 (7.23%)	2 (2.41%)	3 (3.61%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2540	4.15	0.83	924 (36.38%)	1208 (47.56%)	282 (11.10%)	97 (3.82%)	23 (0.91%)	6 (0.24%)

17. The objectives of the course were clearly stated and accomplished.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.47	0.53	39 (48.15%)	41 (50.62%)	1 (1.23%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2536	4.10	0.84	834 (32.89%)	1266 (49.92%)	303 (11.95%)	96 (3.79%)	32 (1.26%)	5 (0.20%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

18. There was a reasonable level of effort expected for the credit hours received.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.28	0.66	30 (37.04%)	46 (56.79%)	3 (3.70%)	2 (2.47%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2536	4.02	0.99	820 (32.33%)	1264 (49.84%)	214 (8.44%)	123 (4.85%)	106 (4.18%)	9 (0.35%)

19. The homework assignments helped me learn the subject matter.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.33	0.65	33 (40.74%)	44 (54.32%)	2 (2.47%)	2 (2.47%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2534	4.10	0.89	841 (33.19%)	1077 (42.50%)	285 (11.25%)	97 (3.83%)	39 (1.54%)	195 (7.70%)

20. The textbook increased my understanding of the material.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
82	3.16	1.18	11 (13.41%)	20 (24.39%)	23 (28.05%)	16 (19.51%)	7 (8.54%)	5 (6.10%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2538	3.48	1.17	414 (16.31%)	782 (30.81%)	468 (18.44%)	270 (10.64%)	163 (6.42%)	441 (17.38%)

21. The course material was well organized and developed.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
82	4.59	0.54	50 (60.98%)	30 (36.59%)	2 (2.44%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2575	4.05	0.95	916 (35.57%)	1128 (43.81%)	329 (12.78%)	140 (5.44%)	56 (2.17%)	6 (0.23%)

22. The instructor was knowledgeable about the subject matter.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
82	4.83	0.38	68 (82.93%)	14 (17.07%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2579	4.50	0.71	1528 (59.25%)	873 (33.85%)	114 (4.42%)	37 (1.43%)	18 (0.70%)	9 (0.35%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

23. The instructor was well prepared for class.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
83	4.78	0.41	65 (78.31%)	18 (21.69%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2578	4.33	0.83	1286 (49.88%)	960 (37.24%)	210 (8.15%)	77 (2.99%)	28 (1.09%)	17 (0.66%)

24. The instructor (not Teaching Assistants) was accessible for individual assistance.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
83	4.42	0.59	37 (44.58%)	38 (45.78%)	4 (4.82%)	0 (0.00%)	0 (0.00%)	4 (4.82%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2583	4.08	0.83	808 (31.28%)	1090 (42.20%)	418 (16.18%)	68 (2.63%)	19 (0.74%)	180 (6.97%)

25. The grading policy was fair.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
83	4.49	0.65	46 (55.42%)	34 (40.96%)	1 (1.20%)	2 (2.41%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2585	3.96	0.95	766 (29.63%)	1205 (46.62%)	368 (14.24%)	169 (6.54%)	56 (2.17%)	21 (0.81%)

26. The instructor responded adequately to in-class questions.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.72	0.45	58 (71.60%)	23 (28.40%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2580	4.23	0.81	1042 (40.39%)	1155 (44.77%)	235 (9.11%)	80 (3.10%)	23 (0.89%)	45 (1.74%)

27. As a teacher, this instructor was better than most others in this School.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
83	4.55	0.65	52 (62.65%)	26 (31.33%)	4 (4.82%)	1 (1.20%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
2586	3.74	1.12	747 (28.89%)	875 (33.84%)	569 (22.00%)	239 (9.24%)	119 (4.60%)	37 (1.43%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

28. The average number of hours per week I spent outside of class preparing for this course was:

~

Question Type: Multiple Choice

~

contributed by Office of the Provost

Results for CS-2110-100

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
83	11 (13.25%)	52 (62.65%)	16 (19.28%)	4 (4.82%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
2546	128 (5.03%)	857 (33.66%)	1123 (44.11%)	294 (11.55%)	144 (5.66%)

29. I learned a great deal in this course.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.36	0.74	40 (48.19%)	36 (43.37%)	4 (4.82%)	3 (3.61%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2531	4.05	0.90	838 (33.11%)	1196 (47.25%)	325 (12.84%)	135 (5.33%)	37 (1.46%)

30. Overall, this was a worthwhile course.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-100

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.59	0.61	53 (63.86%)	27 (32.53%)	2 (2.41%)	1 (1.20%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2537	3.98	0.99	837 (32.99%)	1114 (43.91%)	348 (13.72%)	173 (6.82%)	65 (2.56%)

31. The course's goals and requirements were defined and adhered to by the instructor.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.54	0.53	46 (55.42%)	36 (43.37%)	1 (1.20%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2587	4.18	0.75	893 (34.52%)	1365 (52.76%)	241 (9.32%)	76 (2.94%)	12 (0.46%)

32. The instructor was approachable and made himself/herself available to students outside the classroom.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
82	4.45	0.55	39 (47.56%)	41 (50.00%)	2 (2.44%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2580	4.12	0.81	891 (34.53%)	1204 (46.67%)	401 (15.54%)	66 (2.56%)	18 (0.70%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

33. Overall, the instructor was an effective teacher.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-100, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
83	4.65	0.57	57 (68.67%)	24 (28.92%)	1 (1.20%)	1 (1.20%)	0 (0.00%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2587	4.05	0.97	955 (36.92%)	1065 (41.17%)	358 (13.84%)	148 (5.72%)	61 (2.36%)

34. Please make any overall comments or observations about this course:

~

Question Type: Short Answer

~

contributed by Office of the Provost

Results for CS-2110-100

Total	Individual Answers
42	*See below for Individual Results*

Because of your class I tried Cheerwine...It is now like my favorite soda.

Professor Sherriff, May the 4th be with you. I loved your nerdiness and it seems like you throughly enjoy the subject matter you taught, making it more enjoyable for everyone. Despite getting only getting a 3 on the AP CS B exam, I feel like there wasn't any new material in this course in terms of actual computer science, but there was plenty for computer engineering.

Professor Sherriff has been one of my favorite teachers thus far at UVA. He knows his stuff and makes lectures fun.

none.

I really enjoyed the lectures, though I hope to go deeper into the concepts instead of focusing so much on the process. Lab was a waste of time though. I hoped the TAs teach something instead of just making us work on our own.

its good

Great class. I mentioned a few things above, but you should be able to see those so I won't make you read it twice. It would be cool if you did a lecture or so on GUI design; I wasn't the GUI guy in our group so I never really figured out how it worked (my own fault, but I'm still curious).

Professor Sherriff is a really great Professor. He made the material interesting to me in class!

n/a

Sherriff was a great professor, he was funny, interesting, made the material more enjoyable, and was an effective teacher. Hope to have him again.

Sherriff is the man!

Though I'm not in eschool and do not plan to major/minor in CS, I enjoyed this course. Sherriff is a great teacher, and it's sad he won't be teaching this class next semester

I have to say this! Mr. Sherriff is the best. The course was really interesting and worthwhile!

You're a cool dude Professor Sherriff. I enjoyed taking this course with you!

Fun class and great professor. A reasonable amount of work is expected to do well in the class.

I wasn't required to take this class because I have ap credit, and the vast majority of the java coding stuff I already knew, but I still learned stuff from the other parts of the class, so I stand by my advisors suggestion to take this class

'Twas good.

Sheriff is awesome. I really didn't like the tests, since I found solving the coding questions in 50 minutes was very challenging. I did much better on the homework assignment and particularly enjoyed those. Placing more emphasis on those I think would be fairer.

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.

This course was fun to go to and it was true that Professor Sherriff seemed in a less happier mood during office hours but it was probably because he had much to do.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	He was a bit crabby during his morning office hours...he was in a rush to leave for a lunch meeting???
	Awesome class!!
	Professor Sherriff is awwweeesssooommmeee!!! But in all seriousness, I appreciate the humor that he brings and the way that he uses analogies. Prof Sherriff you might think your random analogies are, well, random, and they are, but they're funny and they always make the topics a lot more clear.
	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.
	I really enjoyed this class and Professor Sherriff's lecutres. However, I was very frustrated with the homework setup. While I understand the real world value of paired programming I don't think using it for every homework assignment as an effective way of learning how to code. With almost every partner I was paired with, they would come to meetings with the whole assignment already coded. This was frustrating because I really looked forward to working through the assignment and learning the material. Because of this, I feel like my coding skills have become handicapped. I suggest having smaller individual weekly assignments that reinforce concepts, then once about every two weeks have a paired assignment.
	He is really really good and entertaining!!!
	sherriff is the man.
	great course!
	good stuff
	Best class, with the best professor, I have taken so far at UVA! I learned a lot and I will be able to take what I learned from the DROID project and actually be able to use it after this course. (Try saying that about something like Calc 2).
	Professor Sherriff took a potentially dull and difficult course and made it fun. Although it was still challenging, I found the course overall very enjoyable.
	I thought that this course was awesome!!! Professor Sherriff did a really good job at explaining the material! He made the subject really fun and really easy to understand. He was SO organized and that REALLY helped me to stay on track and understand the subject better. He was an awesome lecturer and he really knows how to teach the subject so that everyone can understand it. Even though he thinks his analogies are silly, they actually make sense. He was a really good teacher because he actually cared if we understand it. He was really approachable and helpful. I really liked the project because it was fun applying what we had learned to something fun. Professor Sherriff is the one of the best teachers if not the best teacher that I have had thus far at UVa.
	Prof. Sherriff was incredibly well organized.
	My favorite course of the semester
	Great class.
	Good course. I'd suggest it to anyone in the e-school.
	This has been my favorite class at UVA so far, and I expect it to be one of my favorite classes in my four years here. The topics were interesting, and challenging, but not overly hard. Professor Sherriff had interesting lectures and engaged the students every class.
	I hated CS before this class the only reason i had to take it was my major. After taking it when Sherriff he made it fun. He was excited about the course it self and really made me like it. H e didn't just lecture we interacted together. HANDS dwn one of the best teacher in the e school and i am not even Cs major.
	This class has made me consider adding a CS minor onto my already packed double-major schedule. That should say something about the way Sherriff conveys the course material in an easy-to-relate-to and entertaining manner.
	Professor Sherriff was an excellent instructor! I thoroughly enjoyed the class, though I did find it to be quite challenging. One thing I might change would be to have more homework assignments, but that take less time. I felt I could have benefited from more time coding.
	Great teacher and applied concepts to current everyday life.
	The project grading was very un-fair since we had 1 person who did nothing and 1 who barely did anything. Since the groups were chosen, this was very un-fair. Except Prof. Sheriff is a wonderful teacher! Clearly one of the best in UVA! Imran