CS 2110-001 **Software Development Methods** - Fall 2009

ENGR (16333)

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

Respondents: 81 / Enrollment: 90

Summary: CS 2110-001 Software Development Methods - Fall 2009 (16333)

Overall Course Rating

CS-2110-001 Mean 3.92
CS-2110-001 Std Dev 0.97
CS-2110-001 Response Count 403

Difference from Category Mean, Expressed in Category Standard Deviations	-2 -1 0 1 2
	-0.10

SEAS, 2000-level courses Mean 4.01
SEAS, 2000-level courses Std Dev 0.91
SEAS, 2000-level courses Response Count 11659

Overall Instructor Rating

INSTRUCTOR: Sherriff, Mark
Mean 4.52
Std Dev 0.69
Response Count 563

Difference from Category Mean, Expressed in Category Standard Deviations	-2 -1 0 1 2
	0.35

SEAS, 2000-level courses Mean 4.22
SEAS, 2000-level courses Std Dev 0.86
SEAS, 2000-level courses Response Count 17207

~ QUESTIONS AND DETAILS ~

~ ANSWER MATRICES ~

1. 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-2110-001, Sherriff, Mark

Total	0-2 (NA)	3-5 (NA)	6-8 (NA)	9-12 (NA)	13-16 (NA)	17 or more (NA)
81	12 (14.81%)	43 (53.09%)	20 (24.69%)	5 (6.17%)	1 (1.23%)	0 (0.00%)

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)
173	19 (10.98%)	95 (54.91%)	43 (24.86%)	10 (5.78%)	3 (1.73%)	3 (1.73%)

2. 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-2110-001, Sherriff, Mark

Total	Individual Answers
47	*See below for Individual Results*

I thought the project was really good this semester.

yahtzee

none.

I liked the way the project was set up and how the objectives for the project were given at the beginning.

Group projects are a good idea.

Keep us honest.

Maybe more accountability by the TAs.

Regulate how far along each group is better

Allow people to choose their own group. It's true that sometimes in the real world, you have to work with incompetents but I disagree that that is something that should be simulated in this project.

Keep the weekly Raf requirement

teams should be assigned based on major and test grade and not just test grades. some groups had multiple cs majors that were very good at coding while others had students from other majors who did well on tests but could not code well.

No

No

define project roles better outline punishments for no work beforehand

The current format is perfectly fine. But slightly more time should be given.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	less people in the groups - didn't really have need for Q & A Manager
	Some people just suck at working in groups. Great idea to work in groups though, sometimes its hard to get all people to be decent/good
	I have to say I am uncomfortable in rating other people in my group. Granted, I'm not sure of another way to have this project fairly graded, but I feel as though I am generally too nice to grade rate someone in a way that could substantially hurt their grade. I typically give people the benefit of the doubt when it comes to participation, attendance at meetings, or the workload they take on. I think its ridiculous how people only received a multiplier to bring their grade down, and that percentage was not redistributed.
	N/A
	Create project teams by having students rank their programming ability and create teams with a broad range of abilities. You could also create teams based on majors and be sure there is not a team soley of one specific major.
	Do a better job of splitting up good coders, makign the groups more fair. My software architect, who should have been the best, did the least amount of work of anyone in the group.
	I didn't like having the positions within the group because I felt like mine wasn't as important as other positions (I was the testing person). I also felt like some people took over the project and while I wanted to help more, they just did the majority on their own.
	getting 4 different student to not have time conflicts due to classes and exams is quite a challenge
	Make sure the students are actually staying on track - people could BS the RAFs for the weekly checkoffs but actually leave the brunt of the work until the last week
	Spend more class time on how to do project
	Make sure teams have at least one CS/CE major per group
	The project to me was a huge pain, much worse than tests or homeworks. Why not do a scaled down project for two people to do over four weeks? Coordinating 4 different peoples code sucks.
	no
	allow us to plan in greater details
	Not really. The instant messenger project was kind of hard for me even though the code was there for us, we still had to put it together. I guess actually making putting the code together was the hardest part of the project.
	The projects are very helpful. Building an IM client was a great way to work in a larger (than normal) group and over a longer period of time.
	Grouping algorithm is hard to get correct. I understand not letting people select their partners, but I do think it allows students a better chance to receieve the right grade.
	The roles are lopsided, make everyone even
	Emphasize that groups should meet again during the week early on so there is not a ton of programming needed to be done near the end of the project.
	I disliked the voting for a main project. It gave me some false hope of doing something interesting and I really did not like what we ended up doing. I would have preferred having it assigned rather than voting.
	I wish we would work on something more interesting. I know people voted on the project, but I think everyone voted for Instant Messenger because they thought it would be the easiest.
	explain subversion a little bit more thoroughly (ie. the difference between 'Update' and 'Committ,' how to make line-by-line changes/updates to the code with the synchronize panels).
	Have at least one Computer/CS major in a group. While other majors aren't necessarily bad at CS, it would be beneficially I think to have a Computer/CS major per group.
	no project roles
	I realize this is difficult, but students need to be more accurately graded on how much work they do rather than whether or not they have a CS pro in their group. I know people that never wrote a single line of code and spent maybe two hours on the project and still ended up with a 110. I, on the other hand, worked my ass off and got a 95. That is just wrong, especially with the project having so much weight.
	None.
	More interactive and hands on programming assignments, class activities, etc...whenever there is an opportunity to show us code or allow us to follow along, do so

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

I felt the project was an excellent opportunity to learn some more about using java; however, if some of the important topics to coding a larger software system (such as global or static variables) were to be covered earlier in the class, rather than some of the less critical stuff (ie. sorting with collections) some of the work would have been a bit simpler to do.

I enjoyed doing the instant messaging client because it is something I have used before. I would recommend that future projects are somewhat similar in that the students have a familiarity with the project they are using.

Make sure it is interesting. I thought this one was.

I believe that the template used for the project worked well and I don't know that I would change it.

Lab sessions were unproductive, I suggest meetings outside of class.

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

~

Question Type: Multiple Choice

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-001, 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)
81	0 (0.00%)	4 (4.94%)	12 (14.81%)	22 (27.16%)	12 (14.81%)	31 (38.27%)

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)
173	2 (1.16%)	7 (4.05%)	34 (19.65%)	46 (26.59%)	22 (12.72%)	62 (35.84%)

4. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	3.72	0.85	5 (6.25%)	26 (32.50%)	8 (10.00%)	3 (3.75%)	1 (1.25%)	37 (46.25%)

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)
170	3.90	0.86	23 (13.53%)	54 (31.76%)	19 (11.18%)	4 (2.35%)	2 (1.18%)	68 (40.00%)

5. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	3.93	0.96	12 (15.00%)	21 (26.25%)	6 (7.50%)	3 (3.75%)	1 (1.25%)	37 (46.25%)

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)
171	3.98	0.97	34 (19.88%)	46 (26.90%)	17 (9.94%)	5 (2.92%)	3 (1.75%)	66 (38.60%)

6. 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-2110-001, Sherriff, Mark

Total	Individual Answers
60	*See below for Individual Results*

The course was structured very well, most topics seemed relevant. The whole idea of complexity and its usefulness still escapes me....

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I hated the tree lectures because they were not as helpful in the final part of the course when we were asked to compute tree code...It is just a really basic thing that is hard to explain how to use I guess.
	Testing perhaps could have been explained a little more clearly.
	the J unit test just didnt seem that usefull, it seems like everything it did we could have just put the code in the main and get the same results. also, I hate jiggloo.
	none?
	the hacking lecture did not seem relevant to anything I would need
	I didn't like the paired programing. One person usually took over. Also I felt like I couldn't experiment with the program as much, trying different thing just to see how it would work since my partner and I would just be trying to get the homework done as fast as possible. I enjoyed doing the last homework much more and feel like I got a lot more out of it because I got to do it on my own.
	security
	None. I thought they were all very relevant
	I don't think there were any that do not work.
	I will not be coding in the future, so all the long coding homeworks were a real pain.
	Trees.
	OS and internet security
	none
	none
	none
	none
	the coding because that is not an application
	it was pretty much all itneresting / useful
	We didn't really talk as much about networking (Socket class, etc.), though I think it'll be extremely useful going forward...in the future maybe the class could focus on it more or teach it more efficiently
	I wouldnt say that any lectures did not work, but there certainly were a couple dull ones. I imagine you probably know which ones these are and that the topics must be presented as part of the course.
	Threads was probably the weirdest. I get it but I don't see any real practical use within Java although maybe another language.
	I can't think of one
	operating systems, software revisions
	none.
	Agile Development
	None
	None
	None
	None
	None
	None
	None
	The only topic that I didn't really take a lot from was GUI's probably because the learning curve for using jigloo was very steep for me.
	Trees, apparently
	Binary Trees. I still do not understand them.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I didn't really understand the relevance of the recursive tree stuff to the rest of the material in this course. I feel that the amount of emphasis placed on the names of testing methods, diagrams, and development techniques was not as useful as some other information. Though I am sure it plays an important role in the field of computer science, and pure software development those who do not plan on working in pure software do not have very much use for this kind of knowledge and I myself found it difficult to memorize information I had no intent of using. Networking and Internet Security were not covered very well. I liked all lectures. trees N/A The sql injections need more time and guidance. They all seemed useful and worked with each other. I did not like the lectures on UML diagrams, although I think they are pretty important so maybe I just don't like diagrams. Most topics tied in well with one another. Some of the material at the end I didn't see as being as useful. I did not quite fully understand the programming of Trees and recursion. I understood the lecture about the structure of Trees, but did not have enough experience with trees to do very well on the programming questions. Agile Testing The tree/node lecture taught by a random grad student was an EPIC FAIL. Though Professor Sherriff would not want to believe the reason we failed to learn this was simply because of his absence, I firmly stand by the fact we would have succeeded had Sherriff instructed. For each loops, we learned them and rarely used them and the replaced something that was only a couple more characters of code. I still find binomial trees very difficult, but professor Sheriff didn't give that lecture - someone else did. UML All lectures pertained to material on the midterms, thus they were all "useful." Trees and recursion are not my favorite topics. recursion, trees, all others Trees. It seemed like some of the code for trees was pretty arbitrary (ie. tree.left, tree.right). Didn't quite "get" what was going on at first. Some of the higher level topics covered towards the end of year probably won't be too important to students who aren't CS majors, but were fun nonetheless. nothing seemed useless I did not enjoy the guest lecturer who taught on trees. Not because of the subject matter, but because I thought the lecturer was dry and boring. Everything seemed more or less relevant in the long run, possibly with the exception of encryption (which is just a topic of interest, but could be relevant if I get into security)
7. 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-2110-001, Sherriff, Mark Total: 63 Individual Answers: *See below for Individual Results*
	Just the whole systematic side to programming I really enjoy the concept of recursion, I think it can have applications in other subjects and is a very interesting mathematical principle.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I enjoyed the topics on threading and networking, as well as learning some of the features associated with the Java Collections Framework and GUI construction in Java.
	Internet and Security maybe.
	I liked when he had people come in and speak about their programming experiences, the Agile lecture. That was very informative.
	Agile.
	phases of software development lecture`
	n/a
	n/a
	runtime/search algorithms
	The trees seemed really useful.
	Threading
	GUI, recursion, data type
	hacking was cool
	Networking and Maintenance
	Probably threads and networking.
	The different types of lists.
	Lots of them.
	Security
	The lectures on networking and the internet.
	Networks, security, stages of development, maintenence
	General software development/Java coding
	I think the more general lectures on software development and some of the concepts of networking and security will be applicable and useful knowledge no matter where I end up. Also the complexity lectures presented some very enlightening ideas concerning efficiency that can be applied outside of programming.
	Process that goes into development
	probably none - I hope to never code again
	Five phases of development
	Data Structures
	recursion
	Software development methods.
	The software development. I do it on a weekly basis and formal education is good.
	Hard to choose one.
	Because I'm not a CS major, probably just all of the general things about how computer systems work.
	Hacking, I think that by hacking other peoples programs you can find faults that may be in your programs. By doing harmless hacking you realize these defects the easy way instead of the hard way when your program is a failure on the market.
	THE HACKING LECTURE! Okay, no, seriously... the agile development lecture is probably the most pertinent thing that anyone can learn. It's a very effective strategy. Also, the instruction on using Source Control is PIVOTAL for aspiring programmers.
	Using different ADTs, Finding classes from a passage
	Networking
	General coding done throughout the course and review of basic coding principles.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I honestly have no idea, all of them perhaps? How the internet works will prove to be useful in the future The explanation of the waterfall method. UML and networking debgging/JUnit testing hacking I really liked learning about stuff that I will actually use in the future, such as: how the internet works internet security GUI stuff Threads, and general improvement of coding phases of development The early topics on Software Engineering and the phases of development will definitely be useful for me as a systems engineering student. Also, the class/guest lecture on agile development was very familiar to me as it seemed to relate to what I have learned in systems engineering. the subversion seems like somthing veryusefull for later group projects. Probably Collections, threading, and networking Threads I found the networking and maintenance both very useful for the future. The guest lecture. I felt the straight coding/Java topics were must more useful than the overall software development topics. super/sub classes Uhhh.... everything? (currently a CpE major.) I guess if I had to pick one thing: how to use SVN was pretty useful (and supposedly good to know for getting a job). software development Stages of Software Development Blame the User The one in which we discussed processes of development/ functional requirements/ constraints/ etc. not taking another CS course the real world applications such as maintenance, testing, and hacking Event driven programming Threading, Polymorphism, Trees I think the lectures describing the large scale programming process will be the most helpful
8. Which topic/lecture in this course was your favorite and why? ~ Question Type: Short Answer ~ *contributed by Sherriff, Mark (mss2x)*	Results for CS-2110-001, Sherriff, Mark Total: 63 Individual Answers: *See below for Individual Results*
	Internet Security Hacking, obviously. I can't explain why, but most computer-literate kids want to be hackers. They want to be "l33t" I suppose. threads Threads. I liked the last topic on security and the SQL injections. The hacking topic was really cool.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Security because it may be want I want I would like to have a career in.
	I liked talking about the actual software development process, but I also enjoyed talking about Collections and threading
	I really enjoyed the whole class, it's hard to pin a favorite.
	Networking and hacking (basically the last couple weeks of class). It's interesting, and I had always wondered how some of the networking stuff worked/what some of the abbreviations meant. Very useful to know for everyday life. Not that I'll probably be able to actually use the very basic hacking techniques that we learned, but it was cool to see how something like that could be done.
	I liked the part about hacking because it was a lot different from anything else we've seen in CS so far.
	Encryption just because I thought it was very interesting and offered different ways to think.
	when talking about how a program works as a whole and how it is organized; not focusing on the coding at all
	the hacking one, we got to do somthing in class besides just listening.
	GUI - a pain in the butt but very gratifying when done well
	I loved the hacking lecture... the Halloween lecture... the lecture when Sherriff caught someone playing Tetris during the class and delivered him a sick burn... really, all of the lectures! I can't say I was never entertained or failing to learn!
	Hacking, because it would be fun to be good at.
	The lectures about internet security and hacking were definitely the most interesting and enjoyable.
	End of semester when we talked about how the internet works
	Networking - it was the most interesting
	hacking and encrypting because it was cool to learn about
	5 stages of development. I didnt have to write code :-P
	The internet security/hacking topic was my favorite because it felt like you could accomplish something cool on the web.
	Polymorphism, because it is one of the most fundamental concepts of object oriented programming
	gui because i know it the best now and it is actually what i see in every program i run
	I didn't really have a favorite because I struggled with cs throughout the year and didn't enjoy the class.
	Networking was one of my favorites because it explains how da intarnetz workz
	Internet Security because that's what I want to do after I graduate (protect against them).
	Data Structures. I don't know why, I just like data structures.
	internet security
	I enjoyed the lectures on Internet security because it is a very interesting topic and demonstrates the seriousness in investing in secure login information.
	How the internet works, because I use it daily and had NO clue how it works. Now I know and tell everyone.
	Hacking, I think that by hacking other peoples programs you can find faults that may be in your programs. By doing harmless hacking you realize these defects the easy way instead of the hard way when your program is a failure on the market.
	ecryption because it was a different type of coding and didn't have to deal with java
	General overview of OS and security; they were just interesting topics
	My favorite topic was threading.
	Starting to develop multi-class programs.
	The homeworks with the MP3 player because they were fun and less abstract than the other homeworks.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

I enjoyed the networking topic. It made things make more sense about networking in general and programming programs for a network.

I really enjoyed the topic of networking because I found it very interesting.

Encryption - because we got to do a scavenger hunt (& it's a fun topic to begin with).

hacking because i thought it was the most interesting

JCF because it was really useful in HWs

Security, very interesting and different from everything else taught in the class.

I enjoyed the lectures on how the internet works. Its something we use every day, but I feel like very few people understand it.

I enjoyed internet security, I think it presented an interesting perspective on how the internet functions and the danger associated with it.

favorite: in CS? none

hacking. it was awesome.

Complexity. It was the easiest for me.

I especially enjoyed the lectures on internet security.

The semester project was my favorite. It was the most relevant project/topic/whatever-you-want-to-call-it to what we, as students, use on a daily basis.

Hacking

Operating systems

internet security because it was fun learning how to hack

networking, because of the funny way the professor described it

Hacking, because who doesn't want to be able to hack?

I found the information about networking very interesting because it's definitely something I can see myself using in the future.

Maps.

My favorite lecture was probably the one about hacking. Hacking is interesting to me and I found this one was the one that caught my attention the most.

n/a hated almost all

Collections because it was very useful later on.

Threads were the most interesting to me because they are useful and I did not know much about them.

cryptography because the encryption chase was actually fun

9. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.05	0.79	24 (29.63%)	40 (49.38%)	14 (17.28%)	3 (3.70%)	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)
172	4.15	0.79	60 (34.88%)	85 (49.42%)	21 (12.21%)	5 (2.91%)	1 (0.58%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

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

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
80	3.95	0.86	19 (23.75%)	46 (57.50%)	7 (8.75%)	8 (10.00%)	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)
171	4.15	0.79	56 (32.75%)	94 (54.97%)	12 (7.02%)	8 (4.68%)	1 (0.58%)

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

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.98	0.77	17 (20.99%)	51 (62.96%)	7 (8.64%)	6 (7.41%)	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)
173	4.09	0.78	50 (28.90%)	98 (56.65%)	15 (8.67%)	10 (5.78%)	0 (0.00%)

12. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.68	1.37	29 (35.80%)	24 (29.63%)	11 (13.58%)	7 (8.64%)	10 (12.35%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
173	3.72	1.25	59 (34.10%)	50 (28.90%)	33 (19.08%)	18 (10.40%)	13 (7.51%)

13. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
79	4.32	0.74	36 (45.57%)	34 (43.04%)	7 (8.86%)	2 (2.53%)	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)
169	4.34	0.70	77 (45.56%)	74 (43.79%)	16 (9.47%)	2 (1.18%)	0 (0.00%)

14. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.01	1.02	29 (35.80%)	34 (41.98%)	11 (13.58%)	4 (4.94%)	3 (3.70%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
173	4.08	0.92	64 (36.99%)	72 (41.62%)	27 (15.61%)	7 (4.05%)	3 (1.73%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

15. 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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.06	1.35	15 (18.52%)	17 (20.99%)	21 (25.93%)	14 (17.28%)	14 (17.28%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
173	3.27	1.32	39 (22.54%)	40 (23.12%)	45 (26.01%)	26 (15.03%)	23 (13.29%)

16. The subject matter was challenging.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.11	0.78	26 (32.10%)	40 (49.38%)	11 (13.58%)	3 (3.70%)	0 (0.00%)	1 (1.23%)

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)
2338	4.08	0.78	694 (29.68%)	1223 (52.31%)	323 (13.82%)	84 (3.59%)	9 (0.38%)	5 (0.21%)

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-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
79	4.29	0.68	30 (37.97%)	44 (55.70%)	4 (5.06%)	0 (0.00%)	1 (1.27%)	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)
2328	4.17	0.76	789 (33.89%)	1242 (53.35%)	212 (9.11%)	63 (2.71%)	18 (0.77%)	4 (0.17%)

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-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	3.91	0.99	21 (25.93%)	44 (54.32%)	7 (8.64%)	6 (7.41%)	3 (3.70%)	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)
2330	4.03	0.96	746 (32.02%)	1190 (51.07%)	174 (7.47%)	144 (6.18%)	71 (3.05%)	5 (0.21%)

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-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.17	0.83	30 (37.04%)	41 (50.62%)	4 (4.94%)	6 (7.41%)	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)
2329	4.16	0.82	829 (35.59%)	1076 (46.20%)	253 (10.86%)	83 (3.56%)	16 (0.69%)	72 (3.09%)

~ QUESTIONS AND DETAILS ~ | *~ ANSWER MATRICES ~*

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-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	3.10	1.06	7 (8.64%)	21 (25.93%)	27 (33.33%)	17 (20.99%)	5 (6.17%)	4 (4.94%)

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)
2334	3.56	1.10	379 (16.24%)	811 (34.75%)	461 (19.75%)	210 (9.00%)	122 (5.23%)	351 (15.04%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	4.48	0.64	43 (53.75%)	33 (41.25%)	3 (3.75%)	1 (1.25%)	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)
2448	4.14	0.87	914 (37.34%)	1120 (45.75%)	271 (11.07%)	106 (4.33%)	31 (1.27%)	6 (0.25%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.77	0.45	63 (77.78%)	17 (20.99%)	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)
2457	4.59	0.64	1606 (65.36%)	723 (29.43%)	89 (3.62%)	22 (0.90%)	8 (0.33%)	9 (0.37%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.69	0.52	58 (71.60%)	21 (25.93%)	2 (2.47%)	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)
2461	4.41	0.75	1303 (52.95%)	922 (37.46%)	156 (6.34%)	50 (2.03%)	13 (0.53%)	17 (0.69%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.35	0.79	40 (49.38%)	30 (37.04%)	6 (7.41%)	3 (3.70%)	0 (0.00%)	2 (2.47%)

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)
2465	4.18	0.82	935 (37.93%)	972 (39.43%)	342 (13.87%)	60 (2.43%)	15 (0.61%)	141 (5.72%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

25. The grading policy was fair.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	4.25	0.89	35 (43.75%)	37 (46.25%)	3 (3.75%)	3 (3.75%)	2 (2.50%)	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)
2459	3.99	0.92	748 (30.42%)	1153 (46.89%)	336 (13.66%)	157 (6.38%)	40 (1.63%)	25 (1.02%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
79	4.64	0.56	53 (67.09%)	22 (27.85%)	3 (3.80%)	0 (0.00%)	0 (0.00%)	1 (1.27%)

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)
2458	4.31	0.76	1090 (44.34%)	1080 (43.94%)	190 (7.73%)	58 (2.36%)	13 (0.53%)	27 (1.10%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.46	0.76	49 (60.49%)	21 (25.93%)	10 (12.35%)	1 (1.23%)	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)
2459	3.91	1.03	818 (33.27%)	858 (34.89%)	510 (20.74%)	164 (6.67%)	65 (2.64%)	44 (1.79%)

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-2110-001

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
81	7 (8.64%)	44 (54.32%)	25 (30.86%)	5 (6.17%)	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)
2337	154 (6.59%)	858 (36.71%)	914 (39.11%)	291 (12.45%)	120 (5.13%)

29. I learned a great deal in this course.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.09	0.87	27 (33.33%)	41 (50.62%)	6 (7.41%)	7 (8.64%)	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)
2322	4.14	0.83	837 (36.05%)	1114 (47.98%)	259 (11.15%)	92 (3.96%)	20 (0.86%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

30. Overall, this was a worthwhile course.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.09	0.96	31 (38.27%)	35 (43.21%)	7 (8.64%)	7 (8.64%)	1 (1.23%)

Results for SEAS, 2000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2330	4.11	0.91	880 (37.77%)	1036 (44.46%)	251 (10.77%)	126 (5.41%)	37 (1.59%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
80	4.46	0.57	40 (50.00%)	37 (46.25%)	3 (3.75%)	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)
2452	4.23	0.70	882 (35.97%)	1303 (53.14%)	218 (8.89%)	43 (1.75%)	6 (0.24%)

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-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.27	0.77	36 (44.44%)	33 (40.74%)	10 (12.35%)	2 (2.47%)	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)
2458	4.18	0.81	949 (38.61%)	1092 (44.43%)	340 (13.83%)	61 (2.48%)	16 (0.65%)

33. Overall, the instructor was an effective teacher.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-2110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.52	0.67	49 (60.49%)	26 (32.10%)	5 (6.17%)	1 (1.23%)	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)
2469	4.19	0.88	1034 (41.88%)	1030 (41.72%)	276 (11.18%)	95 (3.85%)	34 (1.38%)

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

~

Question Type: Short Answer

~

contributed by Office of the Provost

Results for CS-2110-001

Total	Individual Answers
42	*See below for Individual Results*

Sherriff is the best teacher I've had so far at UVa. Funny, approachable, and keeps things interesting. Teaches in a way that's easy to understand. Makes you want to come to class. Can't say enough good things about him. (Give him a raise! haha. But seriously.) I feel like I learned a ton about CS this semester.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I understand how pair programming can be very effective, especially if both programmers are on the same level, but I think it hurt my learning experience because I allowed my partners to do most of the work (yes that is my fault, but if the homework was individual I would be forced to learn it). Anyways, I do think you did a very good job teaching. Your lectures were interesting and I rarely day-dreamed (although you did catch me playing tetris one time . . . haha). I'm glad I took this class, I learned a lot, and it really broadened my view of computer science.
	This class saved CS in my mind. Bravo and Boo CS101
	This is a respectful course but I did not find it useful for me because I don't think a course like this should be forced upon electrical engineers unless the concentration you choose requires the knowledge of this course.
	really enjoyed the class! Professor Sherriff relates to the students really well, which helps us want to learn and understand the material.
	Professor Sherriff is great
	Sherriff is a great teacher: he knows the material well, presents it well, and makes himself readily available outside of class with office hours. CS is just not a class or subject area I am interested in or will ever actually understand; therefore, my rankings for this being a class where I gained valuable information are low.
	Sherriff was a great teacher and made the subject very enjoyable. Never wanted to not be paying attention in class
	I felt as though pair programming was (with the exception of one team I was in) an opportunity for me to help someone else finish their homework faster than they would have alone, but at a great expense in my time. I felt that the amount of focus on aspects specific to Java does not help students who will enter a workplace which does not make use of Java. While I appreciated learning some "theoretical" CS topics the amount of time which I spent memorizing definitions just to be able to regurgitate them on the test was less than desirable. Though I did learn a good deal about further uses of Java, there was very little generic and applicable knowledge of coding which I took away from this class.
	Few other teachers in this school have reached the upper echelon of tutelage that this man has attained. If I have any disappointment in him, it is that I will not be learning under him in the future. Cheers.
	Sherriff is an amazing teacher; he knows the material, cares that his students understand it, and most importantly, made class ENJOYABLE. As compared to CS 101 (which was hell), Sherriff really make CS much more enjoyable. Kudos to Sherriff. Last thing, why do systems engineers have to take this? My favorite thing from the class was internet stuff, which I can learn in CS 110 without 2 semesters of terrible coding!
	Professor Sherriff is an excellent lecturer
	Great teacher and good class. Hopefully I can have him later when I minor in CS.
	It's easy to fall behind with pair programming if your partner is way better than you, and the project was stressful. Also the grading program was ridiculous, why did so many people get called in after class?
	Great class, couldn't ask for a better professor.
	Needs to be more patient and calm down when students come see him in office hours.
	Hard course but I have learned a lot
	This course was much better than last semester (101E) in many aspects, it was fun to go to class and the homeworks where interesting.
	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
	Good class. I enjoyed it
	Any failings in this course (particularly with regard to the project) were the fault of either myself or "extenuating circumstances." I look forward to taking future CS courses with Professor Sherriff in the future, he has been an engaging and effective instructor.
	I thoroughly enjoyed this class, my only suggestion being that after a while perhaps prof Sherriff should change his intro. Questions, comments,queries gets a little old after a while. Despite this though I would say professor Sherriff is one of the best teachers I've ever had. He is enthusiastic and makes learning enjoyable. His class was good enough that even when i skipped every other class I always wanted to be in CS2110 because I really felt like I was learning and that it was well taught and just all around a good environment. This university is really a better place because of Sherriff and could honestly use more teachers of his caliber.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	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. The professor is humorous and takes time to e-mail students. I did not personally visit his office hours. good course, great teacher. made almost every lecture enjoyable. none Great teacher! I fully enjoyed this course. I learned a lot and look forward to taking my next classes in cs Good teacher. This should be a 4 credit class. This course was quite frustrating at times with difficult assignments and tests, but it was definitely worth it looking back. Thanks for a great semester. the grades were set up in a way that was not acommadating for 1 screw up, i had a few tests that were a's all the labs, and i got almost perfects on every homework except one when i received a 10%, which kinda stinks because i placed alot of time into this course and have a good grasp on the subject matter but my grade does not reflect this. I enjoyed pair programming a lot and I felt with my partner we could always find a good way to do something. However since I didn't actually type all of the code I found some topics that I did not type harder to remember and use during tests. Professor Sherriff was a great professor because he made the subject matter very interesting because of the way he taught it. I am looking forward to taking classes with him in the future. Two things (it was a great course, but suggestions): 1) This a department wide thing, but I don't see how you can have a 1.5 hour lab and make it a 3 credit course. Sherriff did a good job of making that time diminish our outside workload, but this would be 5 credits in CLAS. 4 would be fair. 2) TA's work ethic is terrible (at least for me in the 330 section). They don't care and they don't prepare. I know how hard this is. But it harms the group. Sherriff is a great teacher and a good guy. He was really approachable during class and invited questions, but occasionally I did answer something incorrectly and feel as if he laid the smack down on me. Maybe that was my fault. He could be more inviting at the beginning/end of class and at his office hours; my friends felt he came off somewhat rude at times. Otherwise I loved the class. He's very knowledgeable and fun to learn from. Sherriff is a good teacher who really knows the material. I'm not a big fan of CS, but I enjoyed the class. A very well balanced course, leaves hardly any room for improvement. Great teacher in class. Helped me erase the horrors the CS 101 had etched into my mind. I think I came out of that class knowing nothing, and CS 201 really helped me understand the concepts of computer science. I am a terrible coder who still needed help opening Eclipse at the beginning of November, but i thoroughly enjoyed the HW assignments because I had someone there that could help me through it. While I didnt do most of the coding, the sections I was in charge of were worthwhile and enjoyable with help. I am in systems and can see why this course is required, but I have absolutely no interest in pursuing this field and cringe at the idea of coding so taking this class was a chore for me. You have made it as interesting as possible. I do not feel that you are as kind/approachable/helpful outside of class as you are in class, but maybe i have just come on bad days. Prof. Sherriff does a good job of keeping students interested in class. 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.