CS 1110-002 **Introduction to Programming** - Spring 2012

ENGR (17439)

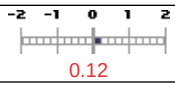
INSTRUCTORS: **Sherriff, Mark (mss2x)**

Respondents: 255 / Enrollment: 297

Summary: CS 1110-002 Introduction to Programming - Spring 2012 (17439)

Overall Course Rating

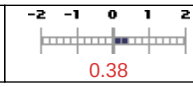
CS-1110-002 Mean 4.07
CS-1110-002 Std Dev 1.04
CS-1110-002 Response Count 1265

Difference from Category Mean, Expressed in Category Standard Deviations	

SEAS, 1000-level courses Mean 3.95
SEAS, 1000-level courses Std Dev 1.02
SEAS, 1000-level courses Response Count 8564

Overall Instructor Rating

INSTRUCTOR: Sherriff, Mark
Mean 4.57
Std Dev 0.63
Response Count 1763

Difference from Category Mean, Expressed in Category Standard Deviations	

SEAS, 1000-level courses Mean 4.23
SEAS, 1000-level courses Std Dev 0.89
SEAS, 1000-level courses Response Count 23390

~ 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
252	3.12	1.35	53 (21.03%)	49 (19.44%)	63 (25.00%)	49 (19.44%)	38 (15.08%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
396	3.11	1.32	76 (19.19%)	83 (20.96%)	99 (25.00%)	83 (20.96%)	55 (13.89%)

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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
253	4.36	0.72	120 (47.43%)	109 (43.08%)	21 (8.30%)	1 (0.40%)	2 (0.79%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
396	4.35	0.75	190 (47.98%)	166 (41.92%)	32 (8.08%)	5 (1.26%)	3 (0.76%)

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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
254	4.47	0.69	142 (55.91%)	94 (37.01%)	14 (5.51%)	3 (1.18%)	1 (0.39%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
397	4.45	0.69	216 (54.41%)	153 (38.54%)	22 (5.54%)	4 (1.01%)	2 (0.50%)

~ 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
254	3.56	1.18	68 (26.77%)	70 (27.56%)	67 (26.38%)	35 (13.78%)	14 (5.51%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
398	3.57	1.20	105 (26.38%)	119 (29.90%)	96 (24.12%)	52 (13.07%)	26 (6.53%)

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

~

Question Type: Short Answer

~

contributed by Sherriff, Mark (mss2x)

Results for CS-1110-002, Sherriff, Mark

Total	Individual Answers
239	*See below for Individual Results*

I liked the intro material, I believe that the simple programming skills that I learned will be able to help me in my life.

Fractals, dude. They're the fiber of the universe.

ArrayList-interesting and have many features to move around with

Loops. Could create display patterns and create more interesting programs without tedious coding

the hello world lecture because it was easy and I knew what I was doing.

Python, easier to understand

Recursion was my favorite topic because it was one of the easiest for me to grasp.

First lecture, the easiest

The lecture on the different searching and sorting algorithms with volunteers at the front of the class. It helped me understand this topic and kept my attention. Also the lecture on the different video game controllers and how they changed over the years was very interesting, as well as the lecture on GPS systems. I also enjoyed the lectures on creating classes, professor Sherriff made these lectures fun and engaging.

recursion the patterns made are beautiful

Loops, it was just fun.

Recursion. Making fractals was a lot of fun!

The lectures about recursion were my favorite because, for once, I actually knew what was happening.

My favorite lecture for this year was when we did the first chase with the Caesar Cipher because it had just the right amount of challenge and it got me out of the classroom.

My favorite topic was learning how to apply loops to make programming easier and more efficient.

I actually have enjoyed learning about recursion, despite it being the hardest topic. My favorite lecture was the one where we edited pictures.

The human computer interactions lecture was very entertaining and interesting. It is an aspect of computer science that I had never noticed or appreciated (before SIS) much.

EVERYTHING

Classes and methods showed me how the basic concepts that we've learned can be scaled up to tremendous size. It showed me how CS (specifically object oriented programming) can be used for real world projects.

My favorite topic was recursion, because it was really cool seeing how much could be done with minimal lines of code.

Didn't care for O.O.P because it was more abstract, enjoyed things like search and rescue where your coding could immediately be registered as correct or incorrect

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The zombie game, because it was fun to be able to use what we'd created, or when he showed us how the game controls had progressed over the years
	Fractals: they are very cool
	creating fractal, we got to have fun and make nice pictures
	Object oriented programming. It let us code a video game.
	OMGZombies
	the lecture on how data is stored in a hard drive was interesting to me because I want to major in EE. I also enjoyed the lecture on human computer interaction, and our intitial discussion into objects and classes.
	The topic where we learned that we have to make things geared towards certain users. I thought it was interesting learning about the hisotry of video game controllers.
	The lecture on human and computer interaction.
	methods and classes. They are interesting.
	Method making was interesting by breaking down big problems into smaller and smaller problems.
	I enjoyed the lectures about the applications of simple programs. It was good to see how code that we could right could be useful in a context outside the class.
	Control of flow because it was like a puzzle
	I really liked the fractals at the end. I think it used the concepts of the class in a fun and creative way. Also I am trying to find a way to use engineering and art together.
	The Chases were a lot of fun while helping to give a better understanding of the material
	Fractals are awesome
	I really enjoyed the lecture on human resources-videogames. It was very entertaining.
	All topics were fun.
	I liked learning about the if statements. I feel like they just made a lot of sense.
	The topic of loop structures and if statements were my favorites because I had not learned them beforehand and I believe they will be very useful in my future as an engineer.
	I enjoyed programming the zombie video game. It was difficult and time consuming, but it was really cool to have a working, interactive program that we created ourselves and were able to customize.
	In homework 5 we programmed a very basic game. I enjoyed that the most. We were able to use everything we had learned so far in the course into a program that was enjoyable to make.
	Programming the video game was my favorite because even though it wasn't very complex, it was cool to say hey look play this game I just created.
	I enjoyed the zombie gaming program the most. It was nice to see the programming put to actual use in the form of a video game.
	Imaging, I thought it was amazing to be able to create something.
	System.out.print("Hello World")
	The HCI video game lecture was the most entertaining. The most interesting topic was learning how to read things off the internet and off files - I didn't realize it was so simple.
	Recursion because I liked the focus on logic over learning syntax.
	My favorite topic was probably loops. I feel like I understood them the best.
	In this course, loops were my favorite. This is because I enjoyed how decision making statements allowed the program to be all to preform functions. However, with loops the program can do even more things.
	Loops because then we were able to make a game.
	Methods - they made programming a lot simpler.
	Fractals! Because they helped me understand recursion and are fun to customize.
	I like the topic of computer/user interraction. It seemed the most relevant to today's world of technology from the user's standpoint.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	recursion. fractals
	I enjoyed the Zombie game and the freedom given.
	The one in which we learned to look at everyday things are try to describe them as objects with fields. It made me thing about things differently.
	I enjoyed making a GUI and incorporating multiple classes in Homework 5.
	Building Classes. I was actually able to write full-blown programs after this topic.
	The cryptography chase was a lot of fun, although it is a hard choice because lectures were so engaging and fun.
	coding games
	Zombie game programming
	classes because making a video game was pretty cool
	Just doing homework, the actual hands on programming is the best.
	Recursion
	I enjoyed learning about fractals and how to create them. The final results always looked so good and on the whole were fun to create. Being able to look at your creation was so rewarding.
	I enjoyed learning about algorithms. They are fun to figure out and serve a very important purpose.
	Class/method interaction
	I really enjoyed the game programming, despite it initially being challenging an frustrating.
	loops because I understand them well.
	loops because they opened up the possibility of using programming to accomplish real tasks
	i honestly really didnt like anything about this course
	I liked some of the applications, especially the recursive methods and fractal drawing, but I most liked using methods
	Array Lists because i have never used them before and i really like how you can manipulate them.
	Fractals....I love fractals!!! So i guess recursion was my favorite.
	I liked everything that had to do with Homework 5 because creating our own Zombie game was the most interesting thing we did in my opinion.
	Lecture on the progression of video game controllers.
	I liked recursion because it made most sense, that is probably because I went to the recitation session that talked about this. Also, making the picture at the end of the year for the last homework assignment was really rewarding! Also, the lecture about game controllers was hilarious and so interesting and fun!
	Recursion drawings
	making zombie game
	Everything about Java.
	User interfaces, presented with fun examples
	Learning a little bit of Python because I got to see another language and also compare computer languages and see how some are easier than others.
	Loops - I felt I understood them best.
	I enjoyed the recursion part of the class because it is like a puzzle and it is challenging.
	I really enjoyed file reading. I know it doesn't sound like the best part, but I've actually been able to use it for my job back home to parse data files.
	bubblesort - seeing people moving around really helped me visualize what was happening
	Classes
	I was a personal fan of switches even though they weren't actually used often in class.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Writing our own classes and creating objects because it was very exciting to be able to program a small game. recursion, it is a cool concept GPS topic- it was very interesting and very practical I really enjoyed learning about how objects worked. For me, it was fun to be able to create actual entites that can be changed, specifically with the zombie game. I don't think I've ever had an assignment that I've enjoyed as much as that one. Fractals because i actually kind of understood it Sherriff's lecture about human computer interaction. He presented the topic in an interesting manner and required audience participation. I very much enjoyed recursion. Even though it is one of the more difficult concepts covered in the class, Professor Sherriff does an excellent job in explaining it. The recursion practice lab. It really challenged me to think in a new way about solving problems. It was difficult. GAME=FUN Overall it was a good intro to programming in general. This isn't necessarily "one" topic, but I really enjoyed the zombies game as well as the fractals. I liked the visual/interactive aspect of it. It was fun to just mess around on it and experiment with various changes. Didn't really have a favorite, most seemed on the same track to me. Recursion because I knew the least about it before. I really enjoyed just learning the basics of programming and exploring the applications of programming. Human Computer Interface - related the programming to things we see everyday. The chases. Those were where students could actually apply material they learned, in addition to combining several topics at once into solving a problem. Zombies Drawing Fractals Printing "Hello, world!" No, just kidding. My favorite topics were IO and methods. I enjoyed the occasional coding free lecture when Sherriff just talked about computers or videogame controllers- it was a nice break. Recursions - didn't know them before Zombies and fractals were pretty cool . . . they were just fun to code. recursion, I kind of understood it. Thats unusual:p didnt have one. Advanced I/O. methods-they were interesting, useful, and fairly easy to learn Can't pick just one, many were very good. fractals because it was a reasonable difficulty The lecture where we talked about GPS and had the demo where students walked around with Prof. Sherriff's I-Phone. It was a really neat demo, it was an interesting topic, and the information seemed really applicable. My favorite topic in this course was loops and methods The Zombie game! it was fun to develop and be more creative. I actually enjoyed the lecture on recursion because it was a new concept to me and I found it fascinating, especially with Professor Sherriff's examples. Additionally I really enjoyed the lectures that were not necessarily coding based - it was nice to learn things that applied to every day life (such as internet connection, GPS, hard-drives, Human-computer interaction, user interface,

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Coding in general, I like this approach to problem solving. It forces you to critically solve problems instead of just going through the steps like in math courses. When we moved over to python in the early part of the course for a bit, for I prefer python in terms of a programming language. The fractal based assignments and lectures were really cool just because I had a background of interest in fractals from a previous teacher I had in high school, so approaching this topic from programming was really cool to me. I really liked programming the zombie game. I personally enjoyed everything, but the game in particular had so many different ways to go about adding different features, it was really fun to work through. loops were my favorite, they made the most sense to me and were so applicable to everything. hello world I didn't really have a favorite topic/ lecture; however I thought most all of the material covered in this course was very useful and relevant. I did prefer lectures where we did not code in class in partners, because I didn't feel like it was productive everytime. python, because it was very similar to java so its pretty easy Fractals cause they were easy and fun i thought it was all interesting I really enjoyed the Chases. But the frisbee lecture really stands out in my mind. That was a topic that I had read about in the book before class and not understood at all. But after the frisbees, oddly enough, it made perfect sense. Recursion because I enjoy the problem solving. Object oriented building because it provided real world applications to the subject Lab was definitely the high point. That with the partnered home works. I liked loops because it allowed me to simplify my code (easily run through files and run my program more smoothly) and I feel its been an important concept in my programs. Sorting lecture - good visual representation of the concepts I enjoyed coding the zombie game, although the assignment was difficult I gained a great appreciation for the work and computer science concepts that make up the game. the loops, they are the easiest to understand My favorite part of this course was the fractals homework and learning about recursion. The chase because it was interactive. I liked the fractals the best, they let us be more creative and weren't as tedious. Recursive Drawing- it was fun and not so code intensive (it did make you think though!) I also liked it because we had some leeway in terms of deciding what we wanted to make. The scavenger hunt around grounds, it was something new and got us out of the class room. Loops, because they are very logical. Figuring our how to solve a problem with programming is logical anyway, but there's something that I especially like about the logic involved with loops. Python recursion I enjoyed the zombie homework because it was really cool to be able to program a game and then actually play it. The things on the search and rescue homework Recursion pictures because they looked pretty cool. I liked when we did the zombie homework seemed like a cool thing we created and the best evidence that we actually did something after hours of work on it zombie game csv files.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	human computer interaction and development - it took a break from nonstop programming, and instead looked at the social implications and adjustments of computers.
	File reading, because it will be useful for easily accessing manipulating data in the future.
	My favorite topic was about if-statements. They are crucial programming and form the foundation for loops as well.
	Recursion: I actually understood was the program was doing and I really enjoyed the fractals HW. My partner and I had a lot of fun making pretty pictures!
	The lecture on user interface was my favorite because it was more applicable and didn't require new information in order to understand it.
	Nothing.
	I really didn't enjoy any topic in this course. I don't like computer science. I'm a fourth year and had to take it for the cognitive science major. I don't think this class should be required for that major.
	My favorite topic was probably loops and if statements. I felt that I understood them a lot and that I could manipulate them freely and easily to solve many of the problems.
	Fractals. It was interesting to learn about programming concepts that could help me to create art.
	Recursion because it was fun to draw fractals
	zombies, It was interesting as well as helpful in learning the material
	Loops. Thought they were really interesting and pretty simple
	My favorite topic was fractals and using turtle because I like drawing and being able to create an image through the computer really interested me.
	if statements and for loops. I liked solving problems and those were easier than recursion
	I personally liked the lecture where we tracked two students on GPS.
	-fractals and recursive methods, because they're fun.
	N/A
	I enjoyed the zombie game homework / topic area, because it allowed us to add in our own features and use our creativity with the program.
	I enjoyed learning all of the material before the second exam. After that, I felt like the lectures weren't quite as clear (it would be helpful if all the code from lectures was posted on Collab rather than only some of it), so my understanding decreased greatly after that.
	Methods and Classes. Learning about how to work with them allowed me to see the practical applications of programming, especially in regards to making games.
	Java
	My favorite topic was our last homework assignment where we drew fractals. I think recursion is interesting, and the algorithms for the fractals were really cool.
	Zombies. It was hard but rewarding.
	The fireworks/zombie game simply because I found them the most interesting. Fractals were cool too.
	Loop statements-this can make a certification process like LEED for Homes so much more simple than evaluating pages and pages of details. Making a program that asks for a specific quality then tells you on the point system what that earns with the rest of your qualities would be SO wonderfully efficient!
	Probably Fractals, I really liked how we could create images and create recursively these intricate designs, had the most benefits after a long bout of programming.
	The GPS homework assignment was really interesting.
	GPS, useful.
	Fractals because it seemed to relate to graphics, which was something I was not expecting from regular CS coding and unaware of.
	The beginning few weeks of the course were my favorite. Working my way through the learning curve that came with Java was really satisfying, and Professor Sheriff and his army TA's were always helpful.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Video Game controllers! GPS lecture- interesting history, fun tracking people Creating my own classes and methods gave me the chance to program on my own instead of using established methods. My favorite topic was making fractals, b/c they were pretty. i liked python because it was simpler loops and recursion because those are so useful I enjoyed learning about while/for loops-the labs and hw were enjoyable! Recursion was my favorite because of its elegance and efficiency. i liked making the zombie game because we could actually do something fun with our homework after it was made. My favorite topic was when we got to classes and objects because we had complex problems and had to tie everything together. I liked recursion because it's so powerful, and when I finally get the code right, it makes me feel like a supergenius. n/a Loops Loops I really enjoyed classes, especially because of the Zombie homework. It was challenging, but it was very fun and forced me to learn the material. It isn't a topic, but I really enjoyed the homework assignments and how they used things we learned in new ways to do something useful... or just fun (fractals). The "chase" lectures were also fun in this way. The lectures where he made analogies with objects helped me understand the material better. Making the Zombie game Writing our own methods and building classes. It makes coding so much easier and allows us to do a lot more. zombies. it was cool making a game gui's because we made a video game reading and writing files becuase it's cool I really enjoyed making the Zombie video game. The Chase I did not have a single favorite topic, I found them all useful. first couple weeks-easier/chill loops loops I liked the major homework assignments because I could see what CS could potentially do. recursion--helped me think of a different, more efficient way to tackle problems. Writing classes and methods in order to model real life objects. Allows one to look at these objects and their specific aspects in the eyes of a programmer. Recursion. It helped me to realize that there is always more than one way to solve a problem, with one likely being more efficient the than other. The beginning because it was so cool seeing code work. using recursion was my favorite part because we actually got to program some pretty cool things and learn more simple ways to program long code. It brought the rest of the class material all together. recursion blows my mind...python is cool

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Recursion - perhaps this is a little nerdy to say but I find it fascinating, especially when using it to create fractals! Recursion. I just liked programming recursive methods. My favorite topic was learning about methods and how you can create your own methods and use them in programs. The transition from just main method programs to building your own methods was mind-blowing. Using recursion to create fractals, it allowed us to not only apply a code to create a program, but to use our creativity to create something that was interesting looking. Recursion. Even though it was probably the most challenging topic, not only is really useful but incredibly interesting. It was completely new to me. My favorite was definitely the short lecture on video games! Because this actually showed how computer science can be applicable in everyday settings and how computer science really is everywhere! Loops were my favorite because more time was spent on teaching them, so I understood them better than other topics. Building objects, it seemed like the most useful concept Fractals learning loops- they are applicsable outside of cs. Classes and Methods Reading and writing files because you can do it in the real world python I personally liked recursion because I understood it and liked the homework assignment with drawing different things. The HCI lecture was a nice diversion, but the lectures in general didn't disappoint often. I really enjoyed the fractals project. It was pretty cool to see how recursion could be applied in code to make fractals and pictures and such. Learning the basics was pretty enjoyable, creating variables, loops, etc. Recurssions because it made reading the program easier My favorite topic was methods because it helped make writing programs a whole lot easier by breaking it down into simpler steps. I enjoyed the section concerning importing online content, as it gave me insight as to how commercial programs may go about integrating online features. The image manipulation in python because i took computer graphics in high school and had done a little of this before
6. 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-1110-002, Sherriff, Mark Total: 228 Individual Answers: *See below for Individual Results*
	Algorithms. Not one topic in particular, but having a basic familiarity with java may prove useful in the future. I know that some understanding of programming will be necessary in engineering. A general understand of computer programming will help more than anything. The logical problem solving will be more beneficial than any code. just knowing the basics of coding Human Computer interaction I think all of them are useful, especially since they are all so interconnected.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	files
	As I am majoring in Computer Science as a result of this course, I believe that the knowledge of loops will be the most valuable topic I have gained from this course.
	The lectures about how real world things use computer science, like GPS units.
	Probably file programming will be the most useful.
	Reading in stuff from a file and sorting it into different types like HW4 and the 10miler question on test 2.
	I'm not really sure any of them will.
	I needed the most help with recursion, so that helped and since I am a Computer Science major, it benefitted me to clear the topic up.
	array list
	I think a basic understanding of algorithms and understanding the programming process was the most important.
	EVERYTHING
	Basics of Programming as well as Human Computer Interface and Advanced IO.
	printing from the scanner to the screen
	I believe recursion will be most useful to my future computer science career.
	Once again, the user interface lecture. I found that lecture intriguing because it was applicable outside the Computer Science world.
	Decision structures will probably be the most useful.
	Oddly enough, the object oriented programming so you can reuse code and such.
	It seems like reading input from files and websites is useful, but of course a strong foundation in loops is good to have.
	The classes and objects lectures. I think these will help with more complex programming.
	basic logic and reasoning.
	Maybe arraylists, since they are applicable to so many things.
	basic coding skills in general
	The topic of loop structures and if statements were my favorites because I had not learned them beforehand and I believe they will be very useful in my future as an engineer.
	The general idea of CS problem solving will be helpful in the future.
	Probably loop methods. They seem like they would have a large number of applications in any area.
	General programming knowledge.
	File I/O
	File I/O
	All of it, especially since I am taking CS 2110 next semester.
	loops because it was used throughout the whole course
	The basic loops will probably be most useful in the future because I understand them so well now.
	All of them are quite useful if I will study computer science later.
	I think methods need to be introduced sooner...it was difficult to wrap my head around methods and classes and actually using them effectively after having programmed for the first time/gotten used to programming without them...
	The entirety
	Recursion
	Recursion

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Recursion
	Recursion
	Recursion because I know it will be useful in the future.
	Learning the basic parts of a program and an algorithm
	I think the ability to make methods will most benefit me in the future.
	I think the most u
	Ferromagnetism?
	Methods and Classes
	Probably recursive method I might even go back and listen to the podcast seems like that will be important for the final.
	Class/method interaction
	File I/O. Needing to analyze large amounts of data and make a representation of that data is huge in any field.
	I believe just the general problem solving skills I used to solve programming problems will be most useful in the future.
	Just the basics of Java; having exposure to it
	Learning how to read and manipulate data for analysis from files is probably the most useful.
	I think in general just problem solving will be what I most apply.
	Looping is , I think, the most important concept.
	Everything about Java.
	The logical way that loops and if-statements work prepared me for problem solving in the future.
	I think file reading and writing. Like I mentioned in the previous answer, I actually found a use for file reading to parse data files in my job back home.
	I think the basic understanding of the fundamentals is the most helpful in the future.
	Object modeling
	Application of other languages
	methods and classes
	Probably loops. I most likely won't be doing any intense coding in the future, however loops are something that are incredibly useful and easy to do. In fact I use them frequently now to cut the time it takes for me to do certain hw questions for another class of mine.
	Anything with recursion since it involves math and thinking about patterns and everything.
	Classes
	Object oriented programming.
	I think the topics of algorithms and problem solving will be the most useful skill in the future. I am planning on majoring in cognitive science and in psychology and in other classes in the major (like cognitive science) algorithms are commonly used to solve problems.
	The most useful lecture was likely the one concerning Python, as it showed how you can incorporate what you have learned from one programming language to another one relatively easily.
	Methods. The concepts behind writing methods are concepts that can be used in all areas of life (i.e. problem solving).
	i have no clue, i dont plan to use CS ever again.
	File reading.
	The small amount of Java I now understand may be useful in the future.
	How complex computers are, hard drive lecture
	I think the topic I'll find the most useful is being able to read and somewhat understand java.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	math functions I don't think that I could name one specific thing that will be most useful to me in the future, but I certainly have a better understanding of what computer science is about and what is involved in it. I think the lecture on loops was the most useful because loops are a basic idea that is used a lot in coding. Even though we really didn't go in depth, the lecture on Human/Computer Interaction will be most useful. I think all the topics were relevant for what I want to do in the future. I am hoping to never have to use computer science again! But the lessons on recursion made me think about problems in a new way which could be helpful in any kind of engineering. advanced IO i dont know..computer programming in general recursion - seems applicable to lots of other things if statements Creating classes Most likely the basics, for and while loops, if statements, Arrays, and file writing. The earliest lectures. Recursion, because it taught how to think in a different way to solve the problem. loops for the same reason Recursive methods. Recursive methods force you to think in a way you are not usually forced to think, so it helps you approach certain problems in general in a different and often more intelligent manner. The lecture on the pig dice game was very useful. loops and algorithms. algorithms Just the general ideas that can be applied to any programming language, for I plan to apply them to C and ruby. The computer science part The whole class, I want to major in computer engineering so java is a good starting language to learn The zombie game-building assignment was a lot of fun and was very good practice in problem-solving and adapting another person's code. all the ones i actually went to The basics of Java and Python programming languages Taking the next CS class all of it will be usefull I think if statements and loops will be the most useful in the future. Also, just problem solving in general will be very useful in the future. methods For loops No idea. Probably just the general problem solving skills, rather than a particular topic. Not so much a single topic, but rather how this class demonstrates how to approach a problem, set up an algorithm and then solve it. When we finally broke into object oriented programing I saw how versatile a programming language could actually be. Recursion and method writing The first one, because it taught me about the importance of the class.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I don't plan on doing much more with programming in the future.
	I think the most useful aspect of this course was not necessarily the programming itself, but learning how to approach problems and solve them in a methodical manner, a technique applicable not only to other programming languages, but engineering in general.
	Everything! Everything was amazing and awesome and will most definitely help me in the future. The best would be algorithmic problem solving though.
	arrays
	All of it.
	recursion
	recursion
	recursion
	I think that File I/O will be the most useful, but I'm planning on majoring in computer engineering, so I think that the entire class will be useful.
	Using scanners.
	I think that the most useful topics were those that reviewed the basic programming language. It is impossible to do anything without understanding the basics and how they are used, so I thought this information was the most beneficial.
	general syntax of java and common programming concepts (loops, if statments, etc)
	Advanced I/O - writing to files is very important
	the problem solving aspect
	I will probably find recursion the most useful.
	The application of computing in everyday life.
	Loops (for,while).
	The early, basic lectures on loops will be most useful. I am a graduating 4th year, so I won't be taking any more CS courses. However, I have already found myself using loops for other purposes--for calculating my own grades, generating tables for my economics classes, among other things.
	N/A
	problem solving in general. How to approach a problem, break it down into manageable pieces then put everything together.
	As also stated above: the lectures that were not necessarily coding based - it was nice to learn things that applied to every day life (such as internet connection, GPS, hard-drives, Human-computer interaction, user interface). I feel these have affected my view of everything related to computers.
	the general problem-solving strategy of breaking down a problem
	Loops, the differentiation between recursive and iterative processes, the caesar decoding. I would love to learn moe about the decryption Sherriff talked about in one of the final lectures.
	Java
	Breaking down problems.
	I think working with classes and the idea of object oriented programming will be most useful.
	Most, if not all in some capacity (CPE major)
	just basic programming knowledge
	General knowledge of java
	Importing files.
	I think that I will find recusion most useful.
	Loops: they apply to problems outside of CS class,
	Probaly the game programming.
	Anything

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I will never program again.
	All of the essential basics to programming that we learned will be most useful because I had never taken a programming class before.
	Breaking big problems into smaller problems.
	None, I don't plan on doing anything with computer science. I may use basic programming skills in mechatronics but very little to none with java.
	The application of CS in the world
	The overall concept of designing a solution to a problem by breaking the problem down into smaller and smaller problems.
	I find I will find all topics useful.
	Programming
	cs in general
	The lectures that I found most useful were the ones about input/output because they taught me how to use more than just what is in Java/Eclipse.
	n/a
	Loops
	Loops
	Loops
	There is not a specific lecture, but learning decision structures, loops, arrays, methods, etc. in Java has already become useful to me outside of CS 1110.
	recursion and looping
	Even though I'm not majoring in CS, I think the problem solving and logic that was required for programming will be helpful in the future.
	I think that I will find having a basic understanding of Java in general will prove to be most useful in the future. It will look good on job applications and may help me in my future career.
	The one's about reading off the internet and using that for applications.
	They were really all great lectures. Professor Sherriff is a great teacher.
	Loops/recursion/all the basic skills we learned in the beginning
	The basic methods
	I think the idea of loops and recursion will be the most useful in the future, because it is almost a way of thinking. You must break a problem down into smaller parts in order to solve it and that can help you solve any sort of problem in the world not just a cs problem.
	debugging programs
	the concept of programming and how it works
	I will probably find the file reading the most useful in the future for I may need to use in when analyzing data.
	Recursion.
	They were all rather helpful and I feel computer science needs a comprehensive understanding of many things.
	java in general
	Classes and methods.
	Either loops or the arrays/arrayLists, both have applications that are important in programming.
	The control structures topic was very fundamental to everything else we learned. I don't know how some classes don't learn that first.
	Everything...but particularly learning how to use basic tools like loops, if statements, and methods - a lot can be accomplished with just those tools.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Learning how to read in files seemed to be the most useful topic. loops loops loops Just knowing the extreme basics of coding and being able to apply said knowledge in the future. Specificcally The HCI lecture. i think just basic understanding of CS that i got from this class will help me in the future but nothing specific. Writing algorithms to solve other problems Writing methods/classes Everything in general. There wasn't a lot in the class that didn't seem useful. Being able to use the basic logic of loops and if statement etc. PROGRAM FORMULAS I think the lectures covering loops will probably be the most useful in the future, because loops are used mainly to solve almost every problem! The base element of coding in java basic rules for java For loops, methods, basic primitive commands - the basics are the whole cornerstone of computer science so learning them I feel I understand how computer's think on a whole new level. Classes and Objects Simple if/then statements and loops are probably the most helpful and the most widely used in any program. Understanding the basis of these is probably the most vital thing to learn in programming. I liked the homework where we plotted markers for the run away family. Fractals because everything else i could have done by hand Using Scanners My god, EVERYTHING! chase Most of the topics will probably be necessary in future CS classes, I suppose the lectures on how to build classes and objects were most useful. I found all parts of the course useful. ArrayList-arrays The basic programming we did at the beginning of the semester programming in general Making classes in Java. user oriented design The class/method lecture loops and array information on how loops work because its applicable to all programming Most all of it. Loops are probably the most useful simply because they have such a wide range of uses in the real world. If I ever need to program again this will most likely be the most important thing to remember Writing our own methods and building classes.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	My favorite was learning to write info to files because its applicable to real life Classes and Methods I think the entire class is very useful, each topic really uses the previous topics to create a whole understanding of computer science, which can be very valuable in the future. recursion, it makes you think I think the GPS lecture really opened my eyes to the potential of computing. Loops on loops on loops on loops. probably recursion and reading files. I think the reading files will be the most usefull I'm not really sure. I feel like the probably methods and classes would be the most useful in the future since I feel that material is the most fundamental to java. fractal drawing Classes and Methods I just feel like that is something we would use a lot in the future. Now that I am a CS major, all of it was very useful to my future. Classes and methods broadened my understanding of computer science and demonstrated the intertwined nature of programs. I think recursion, although arguably the most difficult of the topics, forced students to understand more fully what happens within the program while it runs.
7. 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-1110-002, Sherriff, Mark Total: 222 Individual Answers: *See below for Individual Results*
	Learning about server-human interaction, and file corruption. This is only because my interests do not lie in these fields. I really did not like the array chapter, i found it extremely hard to keep up in class. Arrays. They are confusing. They all can be applicable in some way. The hard drive lecture came off as a bit out of place. Storage could have probably been put in with another lecture to take less time since it was largely overlooked for the final. I do understand the need to know this information though. The "chase" classes did not work too well. I could not figure out the coding fast enough and then the chase was useless because I couldn't do anything without the correct code. recurssion Python..... 1) We spent too little time on it for any of us to claim to be even functionally fluent in it 2) The material we did focus on was entirely foreign - we hadn't even had a chance to do it in Java. Recursion could have used some more time. I feel like the topic of recursion should be explored a bit more in-depth, or at least more related to what was done in the homework. It just seemed like a big jump from what we did in class with rectangles and trees to the complex, dual-recursive fractals that we made in the homework. Could've gone more in depth with the recursion homework in terms of explaining it. I never really learned how specifically it functioned. The python sections will be lost on me in a few days. Either start with both of them from the start or pick one to focus on. The textbook was not necessary; I never bought one and it didn't hurt me at all. The Frisbee pass by reference/pass by value lecture was fun and interactive but I still don't remember the point. Lectures on python and ferromagnetism.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The lectures that didn't have to do with programming did not do anything for me.
	They all were well thought out and executed.
	Python - I would have preferred to focus on Java.I found Python distracting.
	none
	none
	none
	none
	none
	none
	Again, I think all of the lectures were definitely useful!
	I think all parts worked effectively. I would have liked some more historical background on the development of computing and how the components of a computer actually work but in an introductory course I understand that there simply is not enough time to cover everything.
	The hard drive information.
	i didnt really go to many of the lectures.
	sheriff programmed/taught too quickly after the first test
	most of them were useful if you want to major in cs
	recursion blech
	The topic of scope did not seem that much useful to me.
	Hard drive lecture
	I'm probably less likely to use the info about tracing internet pathways than anything else I learned.
	The guest lecture about disk drives seemed to be the least useful in the long run.
	The lecture on hard drives, although moderately interesting, was one I did not consider necessary or useful to me.
	NONE
	I did not think that a zombie game was useful in the longrun; however, it was cool!
	the hard drive lecture
	Searches during class.
	I cannot think of anything presented in class that was useless.
	Python about the chat. It seemed interesting but I was not able to follow it. All of the sudden we just jumped into python.
	Nothing noted.
	All programming we had to do. For me anyway
	Recursion
	Recursion
	Recursions to draw images in java
	The guest lecture at the beginning of the semester on hardware.
	Arrays are very important but they're also very confusing, I would have liked to have worked on those for a longer period of time. The intense details about the mechanisms of a computer that we learned from the guest lecturer weren't as interesting but I think it was important to get one lecture's worth of that.
	nothing
	I thought all lectures were useful.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Not sure
	Nothing in particular. The I/O that we are currently learning are rather cdfi
	the ones i didn't understand (most of them)
	I didn't find anything particularly not useful at all. There were some things I didn't understand as well as others, but nothing that I feel that won't be useful at all.
	Not sure.
	Not sure.
	Some of the lectures got over my head quickly. I learned more in labs than anywhere.
	None, all of the classes appeared practical to me
	the kinda sorta well, almost implication of Python. I can see how it was good to use, but it wasn't as much as I would have expected.
	The random sprinklings of python. I think the course would better be suited to have only one programming language, or at least devote more time to python.
	Chases because not all of us could complete it and the answers were not given and we were not given much opportunities for help or time to complete them.
	I think that the unit on python, although useful in theory, didn't really help me to understand the language at all.
	None of the lectures "didn't work" - sometimes the material was just confusing, but it was usually easy to sort out in lab.
	I did not think that the reading encrypted code was very useful, lecture was confusing and hard to follow and I found myself getting lost.
	method chart.
	the chase
	Arrays seem useless as array lists can always be used.
	learning python along side java
	The picture editing.
	I don't really get why we learned Python, especially since we spent such little time on it.
	I did not find Python very appealing; personally I don't like the fact that we were jumping into the intricacies of a whole different programming language when our main focus was supposed to be on Java.
	the lecture explaining the types of search/sort algorithms. For me this topic took way longer than necessary to explain, especially since the only take aways were the simple complexities.
	PYTHON. maybe if we would have started out with it/ focused more on it, it would have been OK but it has just confused me more than anything.
	The lecture about the workings of the memory system.
	The zombie homework took forever and I will never use it again in my life for anything ever.
	Fractals, while interesting, did not seem to be something that I will use in the long run.
	Many people had problems following along with python, normally due to problems with PyCharm. Maybe spend more time on using PyCharm, so people will be comfortable with it and can pay more attention to the content of the lecture.
	stuff like advance i/o and learning about how hard drives work, etc. things that we did not apply to hw or labs
	I think the python was kind of useless.
	Yoshi's Chase
	Learning Python as a side language, in addition to Java
	Classes
	NA

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The one about storagr on the hard drive.
	Recursive fractals
	Advanced i/o
	Using Python and the server/client lectures.
	Some of the classes about how memory storing works and human computer interaction were not very important, but they did provide a fun and interesting change.
	The chases
	Python. Although it is useful to be familiar with more than one la program, changing from Eclipse to Python can be confusing and not entirely worth it.
	I think all of the topics were useful in the long run.
	The stuff we did towards the end went way over my head.
	Everything seemed to connect in a useful way.
	I think that all of the topics covered will be useful in the long run, maybe just not for me because I do not plan on majoring in CS or getting a job in this field.
	coding in Python
	I would say that probably the lecture about computer disk drives was a little unnecessary. Although I did learn the importance of the word "ferromagnetism" so that's a plus.
	The lecture on ferromagnetism.
	Recursion was interesting, but the applications are beyond me.
	Recursion.
	Most all of the lectures were very useful. I think the lecture covering the hard drives of computers was interesting but never used in the course.
	I think all of the topics would be useful for CS, but not necessarily for my major.
	I have no clue what the frisbee example meant, neither did most of my classmates. It was really entertaining though!
	I enjoyed all the lectures.
	Advanced I/O.
	Recursion does not seem very useful in the long run.
	None.
	The times Professor Sherriff did not lecture, except when Adam Brady subbed in (He was awesome.) tended to be much more boring than the rest of the class, and I didn't really see why learning about the details of a hard drive was useful.
	the chases
	I thought that all of the lectures were useful and interesting.
	The chat client in Python.
	I liked pair programming a lot within the context of the labs. But my experience on the homework was that the effort exerted typically became very lopsided.
	I thought all the topics were useful, but I would have liked to learn more about the importance of writing classes.
	the long lecture series on arrays weren't too useful. I felt they could have been covered directly alongside loops.
	I really did not like the chases. There was not enough time.
	trying to teach some of python and java at the same time, focus on one so we can learn more.
	recursion was really difficult to learn as a large class.
	ferromagnetism!>!?!?!?!?

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	ferromagnatism- hahaha sorry Jay Couldn't really pick one. The chase lectures were no effective. recursions The lecture about disk drives I liked all the lectures, even though i had trouble understanding everything. How hard drives work - it didn't really fit into the course and wasn't taught very well Probably where we are learning images, I think it's too hard for an intro level computer science class and thus since I'm having a hard time grasping the concept I don't think I'll apply it to the future "pass by reference" "pass by value" with the frisbees. Though I appreciate the attempt at diverting from the typical lecture setting, I am still confused as to the use of frisbees, in fact the frisbees may be the reason why I am confused. The disk drive lecture. Python Python Python Python Python Python I thought that learning about Python in the last few lectures was not very useful. We had spent the entire semester in Java getting a grasp of that language when at the last second we changed gears and had to work to understand something different. recursion recursion class, object programming Recursion was a bit confusing. How Computers Remember Stuffs The chases were not as effective as the other lectures because I felt like there was too little time to accomplish them. The coding in Python didn't work for me and I don't find it very useful in the long run. None Fractals because they didn't really teach anything, just added to confusion Whenever we had to code something difficult in lecture, I was usually completely lost along with the people around me. It was too big to get personal attention and help. The chases were the worst. I'm sure they would have been fun if I knew what was going on... The lecture where python was used to create a chat client was really confusing, went too fast, and we were still unfamiliar with python at that point and many of us could not get it installed. Thus, I didn't really learn anything from this lecture. We also had a graduate student teach one lecture about arrays when Sherriff was not there, and I did not find this lecture helpful. The lectures by the TAs were much better than by the grad student. He kept forgetting how to code things and Java and students from the class had to help him correct the syntax, so we didn't really learn much and instead had to keep helping the grad student figure out what his mistakes were. I did not find the chases very useful. I didn't particularly care for Python. It's similar enough to Java that I could probably figure it out on my own if I ever needed to, so I felt learning it was kind of pointless. PyCharm was also really annoying, it ran slowly and only let me use it for 30 minutes when the trial period was up. I don't think there was much in the course that I didn't use. The lecture on the actual computer

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	N/A
	N/A
	There is not a specific one, but sometimes the lecture seems to move really fast and for someone who has never programmed before at all, it can be difficult to follow and understand.
	The Yoshi chase seems too difficult for what it was worth. It'd be better as a homework assignment, or something that I had more time for.
	python. i can't handle java, let alone python.
	randomly throwing in a couple lectures on python.
	Everything seemed to be pretty useful
	I didn't like learning about python
	...
	I thought the "take a few minutes during class" thing to work on a piece of coding was unhelpful. I would prefer a straight up lecture
	Some of the recursion things were foggy in class
	Fractals lecture was a little confusing at the end
	I do not think python will be useful in the long run because I only installed a program that was a free trial for 30 days.
	The chases and class "partner" stuff. I wish there was more lecture during lecture.
	didn't really find anything that was not useful
	Programming involving printing an image to the screen did not seem particularly useful. It felt like we did a lot of work with images but did not actually focus on them during lectures, which felt odd.
	There wasn't anything we learned that I view a not useful, but I suppose if I had to pick something it would be recursion. The only reason I picked recursion is because I don't see myself using it in the future. (It may be the case that I do, but right now I can't really imagine how I could).
	Storage
	I really disliked fractals. I would have rather done something else to practice recursion.
	python maybe.
	The lecture on hard drives was very hard to understand and too much theoretical information.
	The lecture on hard drives and ferromagnetism.
	n/a
	n/a
	n/a
	n/a
	Methods
	Working with binary, hexidecimal, and serialization
	The stuff after fractals
	I did not think the lectures regarding arrays and array lists 'worked'
	Lectures where there was a lot of coding that took place moved way too quickly and it was impossible to understand and follow.
	-
	The course was taught in just a very poor manner. It is not professor Sherriff's fault at all, in fact I found his lectures significantly more interesting then any other lectures I've had since I've been at UVA. This being said the way that computer science is taught here is just inefficient. Putting code up on the screen is just a poor way to learn. I agree that homework should be a big part of the grade but even so there were many times that I would go into the homework with no idea what to do and I would have to teach myself all of it. I am not sure of a fix for this but I think that there are ways to keep people more involved in the lectures and not on Facebook, because just putting code up on the screen and saying memorize this just doesn't work.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Prof. Gurumurthi's lecture on how storage in a computer works.
	Python. Going quickly through it didn't allow me to really understand the material.
	Nothing
	bubble sort
	I still don't have a strong grasp of recursion. I get how to do it, but for many problems wouldn't be able to come up with the recursive case by myself. That being said, I understand why it's useful.
	Nothing to my knowledge.
	python was interesting but for a JAva class it just seemed like a time filler.
	The Advanced I/O lectures
	the hardware aspect
	Just last-minute coding quips.
	could never get the picture extractor to work.
	Class didn't spend enough time on python for it to be worthwhile.
	I honestly can't think of one.
	Many of the "fun" coding classwork assignments were hard to follow and did not work wonderfully.
	python programming. Getting the servers to connect and talk to each other
	At the end, he had lectures on Python, it was hard to follow what he was doing, because lab usually reinforces what he taught, but there wasn't a lab to follow up his lectures on python.
	The ones on advanced I/O, but that could be because I didn't fully understand it.
	The chases. Favored more experienced students, very difficult to complete in a lecture setting.
	The programs that involved GUIs were confusing because GUIs were not discussed in detail during lectures. When we were given code that involved GUI I had a harder time understanding how the GUI was incorporated with the topics that we were covering.
	For the most part I believe most topics were useful, however I think the time spent on python was unnecessary. Core topics such as why its important to know other programming languages, and the differences in different languages were seen earlier on, and examples such as the image manipulation were in my opinion a misuse of time in comparison to how much it was tested on.
	Learning about the parts of the computer.
	The hard drive talk
	I think they are all useful, but I feel like for some of the material we were expected to know things that I did not know from taking previous courses, so I was behind from the beginning.
	Fractals
	Fractals
	python
	python
	python
	Switching between Java and Python was not helpful. It would have been nice to only see work in Java as opposed to bringing up Python every once and a while
	Although fractals were very interesting and more fun to program than other things, I don't think they were nearly as useful as other topics we learned.
	the guest lecture on hard disk drives
	all were pretty useful. I wouldn't say that any of them are not useful in the long run.
	The advanced I/O lectures seemed very strained and squeezed in to small amounts of time, and it was very difficult to understand and follow even though I would say I am a slightly above average CS student, I can only imagine what it was like for everyone else.
	recursion was unclear

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	how hard drives work...lecture for a different class I feel creating images from a file fractals I don't really know, I really doubt I will ever program again.

8. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
252	3.98	0.91	57 (22.62%)	60 (23.81%)	41 (16.27%)	8 (3.17%)	1 (0.40%)	85 (33.73%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
396	3.88	1.01	85 (21.46%)	100 (25.25%)	64 (16.16%)	15 (3.79%)	8 (2.02%)	124 (31.31%)

9. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
254	3.78	0.97	48 (18.90%)	61 (24.02%)	53 (20.87%)	14 (5.51%)	2 (0.79%)	76 (29.92%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
396	3.77	0.99	74 (18.69%)	97 (24.49%)	83 (20.96%)	19 (4.80%)	6 (1.52%)	117 (29.55%)

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

~

Question Type: Multiple Choice

~

contributed by Sherriff, Mark (mss2x)

Results for CS-1110-002, 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)
254	5 (1.97%)	1 (0.39%)	66 (25.98%)	48 (18.90%)	45 (17.72%)	89 (35.04%)

Results for SEAS, 1000-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)
398	5 (1.26%)	3 (0.75%)	102 (25.63%)	81 (20.35%)	65 (16.33%)	142 (35.68%)

11. How would you rate the availability of TAs?

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
251	2.99	0.85	74 (29.48%)	115 (45.82%)	50 (19.92%)	10 (3.98%)	2 (0.80%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
393	3.04	0.84	126 (32.06%)	174 (44.27%)	76 (19.34%)	15 (3.82%)	2 (0.51%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

12. How would you rate the helpfulness of the TAs?

~

Question Type: Likert

~

contributed by Sherriff, Mark (mss2x)

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
251	3.05	0.81	74 (29.48%)	128 (51.00%)	39 (15.54%)	8 (3.19%)	2 (0.80%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
395	3.04	0.81	113 (28.61%)	204 (51.65%)	63 (15.95%)	11 (2.78%)	4 (1.01%)

13. How often did you make use of the TA office hours?

~

Question Type: Multiple Choice

~

contributed by Sherriff, Mark (mss2x)

Results for CS-1110-002, Sherriff, Mark

Total	Every week (NA)	Every other week (NA)	Once per assignment (NA)	Rarely (NA)	Never (NA)
253	32 (12.65%)	41 (16.21%)	63 (24.90%)	74 (29.25%)	43 (17.00%)

Results for SEAS, 1000-level courses

Total	Every week (NA)	Every other week (NA)	Once per assignment (NA)	Rarely (NA)	Never (NA)
396	44 (11.11%)	79 (19.95%)	86 (21.72%)	112 (28.28%)	75 (18.94%)

14. Any specific comments about the TAs you would like to share?

~

Question Type: Short Answer

~

contributed by Sherriff, Mark (mss2x)

Results for CS-1110-002, Sherriff, Mark

Total	Individual Answers
149	*See below for Individual Results*

I really appreciate them being there during lecture to help out with in-class assignments.

some TA are harsh with grading some are not, this difference more or less result in unfairness

The TAs were very helpful and were even will to help with a quick question when not in office hours. Also Dan is awesome.

TA mights need to tell us why we code in this way rather how we code.

army of TA's but too slow

Kyle is extremely helpful.

sometimes they were incredibly rude. I understand that when students ask very vague questions it's frustrating and some of them are my age but they are still seen as teachers in some aspect and it's very demeaning and disrespectful to respond angrily to a good question simply because other students asked vague questions

Hunter thinks he's the man and bragging about how you are such a good athlete, broke so many bones, and have done so many things in your life while trying to hit on another TA when she is teaching lab is just plain annoying. Every Thursday lab at 12:30 Hunter would be there hitting on her and distracting her and all the students in the class.

Most of the TAs were very helpful, but a few could be quite condescending at times. Also, the office hours could get very crowded at times and it would take hours to get any help, and the help would be very short.

I heard that they answered questions to generally.

They were knowledgable and personable. Thanks.

Some were definitely more helpful than others. I found the older the TA (meaning the higher the year), the more helpful. I also never went to office hours until the last 3 HW assignments.

They are always happy to help and are very approachable. Also the fact that there are office hours almost every hour of the day was very helpful.

More TA times.

They are helpful; however, sometimes after waiting a long time just to get their advice they would only give me one tip then go away.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	they were very helpful
	It was very helpful when TAs were giving hints about where to start with the homework, they did a very good job
	no
	no
	none
	none
	Both Kevins are excellent TA's.
	the TA's in lab section 102 were AMAZING. so helpful, i really couldnt have made it through this class without them.
	They were very helpful
	They dedication and patience was very admirable!
	They were helpful but there was usually and unmanageable line to get help
	They were very helpful and helped understand the homework assignments.
	It would be nice if there were more TAs there, especially when we have difficult homework assignments due. It is okay for TAs to be first years or other students who have taken the class before and done well in it, they don't have to be TAs but it is super helpful having them be TAs because they just take one glance at the program and can tell what is wrong. They are soooo helpful! Thank you!!!!!!!
	They were really available and also helped me through all of the HW assignments.
	Very friendly!
	They generally seemed to know what they were doing and were helpful with the assignments.
	They were very helpful and enjoyable to be around. They did not view us as a burden and truly cared about our learning.
	If it wasn't for the TA's and office hours there is no way I would be passing this course.
	Kevin is the best, he helped me the most and helped me understand each program step by step so that I had a clear idea of what I doing. Hunter was also very helpful.
	They all were very friendly and knowledgeable and seemed eager to help us understand the material.
	Not really
	Some of the tas were very helpful, especially Casey and Erin.
	They were nice and sometimes helpful
	It's kind of crazy to have to sit in Thornton for an hour to receive help. Even on a Sunday or Monday before an assignment was due on Wednesday
	helpful when needed
	He was funny and helped out
	I found that most of the TAs were very helpful and knowledgable. However, I found it very frustrating to get an available TAs during some of the later homework assignments. Although there are many office hours, during the office hours at night I found that I had to wait sometimes up to 2 hours to talk to one TA only for about 5 minutes. So if you have multiple problems with the code it takes a very long time to get help.
	None
	TAs were very helpful. responded quickly to regrade requests and were really helpful in homework assignments
	Most of them are really helpful but not in the sense that they just give you the answers - they just give you a hint and push you in the right direction of how to think about your problem.
	Some were better than others (Hunter was the best)
	They were helpful and knowledgeable, and answered my questions on both specific and conceptual levels.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	very chill, but helpful at the same time
	No
	No
	nope
	nope
	Sometimes I felt stupid asking certain questions, so I was intimidated when it came to approaching some of them. Casey was really great though.
	Kevin Whelan is a BOSS HOG!!!!
	Kyle Ames was fantastic. He was really good at explaining the material in a way that I could understand.
	N/A
	N/A
	nice people
	I'm still not sure if they really go through the list (where students put their names down for help)in order because I was skipped sometimes.
	The TAs were the only helpful part of this course. Without them teaching the labs and explaining the concepts, I would be completely lost. Lectures were useless for me.
	Several times the TAs cancelled their office hours, but didn't remove them on the calendar. This was especially true over the weekend; I went 3 times and there was never a TA there.
	They are generally very helpful. However it would be helpful if they were given clearer instructions on the assignments, because sometimes they would contradict each other and it could be confusing.
	Courtney Maimon was by far the most helpful!
	During office hours in stacks, most of the TAs didn't follow the schedule or kept going back to the same person, even though there was a list. Favoritism obvious had a big role in this; more than a few time I spent hours waiting for a TA would would only get through one or two people in that time frame.
	Hunter is not very nice.
	The TAs were very helpful at their office hours and in lab, and they were a great resource for anything related to CS. Specifically, Dan Epstein was consistently the only TA in lecture and always willing to help.
	One of my lab TAs was very difficult to understand
	Never thought they were very helpful.
	They were very helpful.
	They were very helpful.
	Kyle was awesome.
	They were very helpful, but at peak hours could be super busy and required a long wait.
	Peter Sahajian rocked.
	Not particularly
	Sometimes in lab gave way to long answers to simple questions. I guess they thought I was dumb.
	They were very helpful when I needed help on homework, but I would often have to wait hours for a TA to finally reach my name on the list.
	The TA's always did their best to help everyone who asked, but at times there simply wasn't enough time for them to get to everyone.
	Jason is awesome.
	Some TAs seemed a tad bitter about their office hours, not all though.
	n/a

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	They were usually nice and friendly, at times they spoke over my head because i was lost a lot and couldnt really communicate with them because i was clueless
	They were helpful if you had a small question or a yes or no question, but otherwise not so much. Once on recursion, when I was having difficulty grasping it, my TA suggested I write out the problem in pseudo-code. A great suggestion, so I did this. After 30 minutes of waiting, my TA finally came back, to which I showed my pseudo-code. He said it was right, and then told me to convert this to actual code. When I said I wasn't quite sure how, he told me to write out the pseudo-code again. As you can imagine, this did little to help me understand how to code it into java. While I understand that the TA should not simply give you the answer, it would have been quite helpful for him to steered me in the right direction on this particular problem, so I could have one example to go off of. I now understand it, but only after looking online and in the book.
	Some were helpful, but some were too vague in their help, especially for someone who is completely new to computer science like I am
	Sometimes different TA's would say different things about what would be expected for the grading which was confusing sometimes.
	Ivan and Kyle are cool dudes
	They were very knowledgable and helpful.
	They're all very nice and understanding, especially when you make stupid mistakes.
	Kevin McVey was awesome! He was so helpful every time I went. Casey Huang, Dan Epstein, and Courtney Maimon were also very helpful
	Some TAs acted as if they didn't want to help, once I even got a "Well, I could sit here and help you through it but I don't have the time so keep working" that was not helpful at all.
	All the Wednesday night TAs were the best: Hunter, Adam, and both Daniels. But let Hunter know that he needs to grade more leniently on tests. Just a little though :)
	TA's all have different ways of solving each problem so its difficult to talk to more than one on the same program. It would be better if they knew one way we should be solving it.
	It was frustrating when TA wouldn't show up to their TA office hours, without any notice and then I would sit there for an hour or so seeing if they would show up or if the next TA would show up and then the next TA wouldn't show up either. But overall, I really appreciated their help.
	The TAs that I interacted with were very knowledgeable, and I really liked how excited they got about stuff.
	The TA's that ran my lab (109) were always really helpful and patient.
	Erin (from lab 104) is SOOOO helpful. She doesn't just throw the answer at you like some other TA's might just so they can save time and move on to another person- she actually provides hints and helpful feedback so that you figure it out on your own
	Almost all of the TA's I dealt with were extremely arrogant and almost offensive when you tried to come to them for help. Many of the TA's seemed to revel in the fact that they knew more than you about computer programming, and were extremely disrespectful and unprofessional to myself and many of my classmates. On top of that it seemed many of the guy TA's were quite eager to help out girls and perfectly "nice" to them, but if a guy asked them something they would make no attempt to hide the fact that they thought you were an idiot and wasting their time. This behavior was disgustingly obvious to the point where I would have to go to the help sessions with a good looking girl if I actually wanted to get any help that day. Everyone I talked to in my lab and lecture had similar complaints about the TA's... it was very apparent that this was a huge ego trip for many of them.
	It was annoying when they didn't show up and sometimes it seemed like there could be more of them. Also it would be nice if they would show up on Thursday afternoons. Marco was a jerk to everyone but some of them were really great and super helpful.
	they could be more specific and helpful instead of raising back questions
	GOOD
	The TA's were nice--I didn't really interact much with them.
	Sometimes it was difficult to get ahold of the TAs just because the line was so long at office hours and then when the time was up sometimes they would leave without telling anyone so we'd be sitting there waiting for the TA and it took a minute to realize they'd left. I understand they have time limits, but if you're sitting there specifically waiting for a TA and they leave without telling us it makes it difficult because we're stuck sitting there waiting for them and they aren't coming. I have talked with other students who shared the same concern.
	i like the funny ones
	No.
	No.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	They were helpful.
	They need to understand the learning curve at the beginning for students who have not had the opportunity to program before. I felt looked down upon by some TA's.
	Nope
	Nope
	I thought many were good, but some could do a better job of not seeming so condescending. We are very new (most o us) to the language of Java, and it does not come as easily to some people.
	Some TAs strongly favored their friends. One went directly to help his friend even though there had been many students in line waiting to get help. Some did not have full knowledge of the homework assignment. Some would merely refer to the professor's key and say that's that because it says so in the key rather than being able to explain it.
	Kyle and Ivan are great TAs!
	There were very obviously good TAs to go to and NotSoGood TAs to go to. I mean, everyone obviously knew their stuff but some were a lot better at explaining it than others.
	Some didn't show up when they were supposed to
	I wish they would explain things better at the beginning of lab sections
	They were pretty helpful whenever questions were asked during lab.
	SO NICE and helpful
	More TAs are needed at a single time so one doesn't have to wait around for an hour and a half before a TA can help them.
	TA's (especially in my lab group 107) were, in general very helpful. My 107 TA's really helped me understand recursion much better.
	Although some TAs were really helpful, others didn't take the time to understand what you were asking. As a solution to your problem they sometimes just gave the "the way I would do it" response. ALthough this might get you through the assignment (and this is not even true all the times because sometimes you algorithm is just incompatible with theirs) you are not getting your question answered. You might finish your assignment the way they would do it, and still have no clue of what is going on. There was one TA who really was able to answer all my questions, and took the time to understand what I was actually asking: Courtney
	The TAs in the 8 am lab were really helpful the whole semester! Even though it was early in the morning they still made us laugh and helped wake us up!
	Peter was great at reviewing the materials from lecture and solidifying the concepts, breaking down the concepts into easy to understand descriptions and examples.
	Dan was really cool. Marko was a bit mean at times. He didn't have the patience for us when we were clueless.
	It was confusing when one TA would tell us to do something, and another TA would tell us that that was wrong and to do something else.
	The only time I used them was during labs.
	Their enthusiasm needs to be more contagious.
	Hunter is boisterous.
	Good
	Some TA's were excellent and some seemed like they had a lack of understanding of the assignment.
	Sometimes they seemed condescending, granted this is very basic material for them but its hard when you're new to it
	Hunter and Dan are particularly awesome. For the most part the TA's were really helpful, even though you know they probably think you're an idiot for not understanding basic things. I liked how they would help you as long as you needed in stacks and not rush you.
	I did not frequently attend office hours because the first time I went it took nearly an hour to speak to a TA.
	Kevin was extremely helpful in lab and Dan Epstein was very friendly and available.
	TA's vary greatly in helpfulness and quality. I encountered a TA during office hours that was very friendly and helpful while sometimes my lab TA's would be dismissive or unhelpful.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

With the exception of a particular few, all TAs were very helpful, especially during stressful homework crunch times.

TAs are very helpful.

Some of them are fantastic. Others can be a little sassy, which isn't very helpful when you're legitimately confused.

None.

None.

None.

adam-attitude, condescending

They are pretty cool, and mostly glad to help.

As a whole, I thought the TA's were very knowledgeable and they were very eager to help us out.

The TA's were very helpful, but it was difficult to get ample time with them, mostly due to the incredible number of students trying to access them.

some of them were cocky and annoying

Daniel Epstein is the man!

I thought they were great. They were really helpful, and they really helped, especially on the homework. They were also really helpful because they would make sure I had a clear understanding of the concepts before they moved on to help someone else. I'm also aware that a lot of TAs often stayed well beyond their scheduled hours, which was very considerate of them.

they were good in terms of helpfulness, but sometimes their availability was so limited. I once had to wait an hour and a half before I was helped by one at office hours

The TA's were awesome!

15. The course addressed technically rigorous subject matter consistent with the course objectives.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
251	4.36	0.65	113 (45.02%)	115 (45.82%)	21 (8.37%)	1 (0.40%)	0 (0.00%)	1 (0.40%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
1713	4.18	0.83	654 (38.18%)	809 (47.23%)	169 (9.87%)	50 (2.92%)	24 (1.40%)	7 (0.41%)

16. The instructor used methods other than/in addition to traditional lectures (for example, active learning, in-class problems, collaborative learning, in-class discussion) effectively in this course.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
253	4.57	0.64	162 (64.03%)	75 (29.64%)	14 (5.53%)	2 (0.79%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3373	4.17	0.96	1018 (30.18%)	798 (23.66%)	308 (9.13%)	81 (2.40%)	53 (1.57%)	1115 (33.06%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

17. 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-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
254	4.07	1.03	102 (40.16%)	101 (39.76%)	25 (9.84%)	18 (7.09%)	8 (3.15%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
1718	4.12	0.90	632 (36.79%)	797 (46.39%)	173 (10.07%)	82 (4.77%)	30 (1.75%)	4 (0.23%)

18. 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-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
253	4.53	0.68	155 (61.26%)	81 (32.02%)	14 (5.53%)	2 (0.79%)	1 (0.40%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
1715	4.10	0.99	687 (40.06%)	661 (38.54%)	184 (10.73%)	106 (6.18%)	41 (2.39%)	36 (2.10%)

19. 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-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
253	3.76	1.20	81 (32.02%)	83 (32.81%)	42 (16.60%)	25 (9.88%)	16 (6.32%)	6 (2.37%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
1714	3.61	1.14	362 (21.12%)	650 (37.92%)	333 (19.43%)	171 (9.98%)	107 (6.24%)	91 (5.31%)

20. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
252	4.35	0.73	122 (48.41%)	102 (40.48%)	23 (9.13%)	5 (1.98%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3356	4.11	0.91	860 (25.63%)	929 (27.68%)	317 (9.45%)	89 (2.65%)	37 (1.10%)	1124 (33.49%)

21. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
250	4.78	0.49	201 (80.40%)	43 (17.20%)	5 (2.00%)	1 (0.40%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3329	4.45	0.78	1314 (39.47%)	633 (19.01%)	219 (6.58%)	32 (0.96%)	14 (0.42%)	1117 (33.55%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

22. The instructor was well prepared for class.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
251	4.73	0.49	187 (74.50%)	59 (23.51%)	5 (1.99%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3341	4.36	0.79	1162 (34.78%)	750 (22.45%)	259 (7.75%)	32 (0.96%)	14 (0.42%)	1124 (33.64%)

23. I received adequate preparation from the prior courses in the curriculum to be successful in this course.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
254	3.38	1.21	37 (14.57%)	30 (11.81%)	57 (22.44%)	19 (7.48%)	13 (5.12%)	98 (38.58%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
1704	3.63	1.12	288 (16.90%)	346 (20.31%)	330 (19.37%)	93 (5.46%)	63 (3.70%)	584 (34.27%)

24. The grading policy was fair.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
254	4.30	0.74	113 (44.49%)	111 (43.70%)	24 (9.45%)	6 (2.36%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3339	4.03	0.92	743 (22.25%)	905 (27.10%)	375 (11.23%)	113 (3.38%)	29 (0.87%)	1174 (35.16%)

25. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
252	4.59	0.60	161 (63.89%)	77 (30.56%)	12 (4.76%)	1 (0.40%)	0 (0.00%)	1 (0.40%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3324	4.31	0.87	1144 (34.42%)	687 (20.67%)	270 (8.12%)	54 (1.62%)	27 (0.81%)	1142 (34.36%)

26. The instructor effectively used technology in support of the learning goals for this course.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
251	4.71	0.51	183 (72.91%)	62 (24.70%)	6 (2.39%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
3328	4.19	0.90	962 (28.91%)	787 (23.65%)	338 (10.16%)	58 (1.74%)	34 (1.02%)	1149 (34.53%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

27. 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-1110-002

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
254	3 (1.18%)	96 (37.80%)	117 (46.06%)	26 (10.24%)	12 (4.72%)

Results for SEAS, 1000-level courses

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
1715	141 (8.22%)	794 (46.30%)	615 (35.86%)	117 (6.82%)	48 (2.80%)

28. I learned a great deal in this course.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
253	4.40	0.76	134 (52.96%)	91 (35.97%)	24 (9.49%)	2 (0.79%)	2 (0.79%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
1712	3.97	1.08	642 (37.50%)	630 (36.80%)	256 (14.95%)	112 (6.54%)	72 (4.21%)

29. Overall, this was a worthwhile course.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-1110-002

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
253	4.35	0.85	136 (53.75%)	81 (32.02%)	30 (11.86%)	1 (0.40%)	5 (1.98%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
1710	3.88	1.20	648 (37.89%)	583 (34.09%)	224 (13.10%)	132 (7.72%)	123 (7.19%)

30. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
252	4.52	0.57	140 (55.56%)	103 (40.87%)	9 (3.57%)	0 (0.00%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
3260	3.76	1.06	938 (28.77%)	1000 (30.67%)	1095 (33.59%)	42 (1.29%)	185 (5.67%)

31. 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-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
254	4.13	0.88	99 (38.98%)	105 (41.34%)	38 (14.96%)	9 (3.54%)	3 (1.18%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
3244	3.72	1.04	951 (29.32%)	782 (24.11%)	1288 (39.70%)	97 (2.99%)	126 (3.88%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

32. Overall, the instructor was an effective teacher.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-1110-002, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
254	4.56	0.65	163 (64.17%)	73 (28.74%)	16 (6.30%)	2 (0.79%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
3244	3.80	1.06	1083 (33.38%)	768 (23.67%)	1181 (36.41%)	82 (2.53%)	130 (4.01%)

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

~

Question Type: Short Answer

~

contributed by Office of the Provost

Results for CS-1110-002

Total	Individual Answers
148	*See below for Individual Results*

this was a good helpful course ! thanks

Overall, I thoroughly enjoyed this course. I felt at the beginning it moved a little slow but made up for it with recursion/python/advanced io topics. I also would not recommend switching the class to Python just because I feel like you learn the programming process better in Java than in Python

Fair class

I think that I could have learned more from this course if I had put more effort in. However, I'm a fourth year and I have little need for computing in my future career goal(med school). I think this course is great for the engineering students in the class.

The homeworks took far too long, I think we could learn just as much on an assignment more similar to what we dealt with in lab. They often made me feel disadvantaged as someone who knew little CS before the class and wasn't gifted in programming. Sitting in TA office hours for over ten hours a week to get guidance wasn't enjoyable.

Loved Sherriff. Really made class enjoyable and was an effective yet still entertaining teacher. I was worried I'd hate the course coming in, but found I kinda enjoyed a lot of the aspects of computer science, though not all of them

Sherriff was super awesome. Every lecture was fun and very informative. However, sometimes, things would go too fast for me and I would be lost for the rest of the lecture. CS isn't something I'm very interested in, but this class definitely taught me a lot, and was fun nonetheless.

Good class

Although I was not particularly fond of computer science, the TA's and especially Sherriff helped to make the course significantly more enjoyable.

Sherriff is a great person, and a very good teacher.

It was fun to learn new programming language.

Sometimes, he goes a little fast, and I understand that it is a giant lecture, but the review sessions and TAs were readily available, so overall, it was a good class.

Most of the codes done in class were posted online but I feel like more example codes from class would help me to understand the subject.

Mark Sherriff is the man!

The course was extremely effective in making me learn how to program and understand computers. The only thing that I didn't necessarily enjoy about the course was that it moved a bit too fast for someone who didn't know anything about programming. We moved at a good pace in the beginning of the semester but by the time homework three was handed out I got really lost and struggled to really understand the material and complete the homework and felt that I fell behind compared to most of the class. The homework's were extremely helpful in helping me learn the material, I just wish I felt better prepared to tackle the homework assignments.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Although this class is meant for people with zero programming experience, I think it should be advertised as a fast course because for people like me who have never done any programming before in their lives, it was very difficult and I always felt stupid and felt like everyone was better than me because many students had taken a programming class before. I think it is best to separate students based on their level of programming so that they dumb ones don't feel dumb. The 1112 class should be more highly advertised because I regret not having taken that one even though Sherriff is an AMAZINGLY effective and funny professor!!!!!!!ONE!!!!!!!!!
	the quizzes were dumb and seemed like a waste of time and a hindrance to my grade simply because I forgot to do half of them
	I learned a great deal in this course, but think some topics could have been more clearly taught and spent more time on.
	I enjoyed the course. I learned Python in sophomore year of high school but forgot most of the material, so it was interesting to learn a new programming language. It encouraged me to go back and refresh my skills with Python, especially after programming basic video games and other things.
	The material was challenging, but I really enjoyed it!
	IT WAS GREAT! I really hope I have Sherriff again as a teacher!!
	Great Class
	Good and stuff
	It has been good to get exposure with computer programming. I do not wish to pursue computer science further. That is in no way a reflection of this organization or execution of this course.
	Very fun class to take because at day 1 you don't even know how to use java to print anything to the screen but by the last day of the semester you are simply amazed at everything you learned. It can be hard at times, but the book is very helpful as well as the TA office hours. Definitely a class worth taking even if it is only to give you some background in a programming language.
	Mark Sherriff is awesome. Nerds are awesome. Probably my favorite instructor/class this semester.
	I didn't know anything about Computer Science before taking this class but I felt well prepared for the tests. The weekly quizzes were somewhat of a pain because if you missed one then you would get an 80% because there were only 5 questions but since they were worth very little it didn't really make much of a difference. All in all, the only reason I did well in this class is because the TA's and Professor Sheriff taught the course very well!
	Professor Sherriff is a really cool guy, but I cannot follow his lectures. He begins them fine, but when he starts coding, he codes incredibly fast and doesn't always explain what he is doing. Sometimes the lecture code is posted, but it doesn't happen every time. I went to his office hours, and he was incredibly unhelpful. He rolled his eyes at me and made a TA answer my question (which wasn't even unintelligent). He makes homework assignments that contain information or portions of code that we have never learned before, such as for the Credit Card Assignment.
	I really enjoyed having Prof Sherriff as a teacher. He did a great job teaching me comp sci.
	Sherriff really is a great professor for the intro CS class because he makes things fun and easy to understand.
	Fantastic course, entertaining and engaging! Prof Sherriff is an excellent teacher for the course!
	Sherriff is great. This class is difficult for people without prior experience in a lecture format. Written tests are difficult when all the experience is on a computer in class and for homework.
	I like this class.
	Coming into CS I had no idea what to expect. Within the first lecture I was already surprised by how much I liked it/how interesting it was. I thought Professor Sherriff did a great job of making class interactive, fun, interesting, and still very informative.
	Sometimes the homework was a bit much, and other times there was no homework, so sometimes it was hard to plan my time/schedule because of this.
	I think it is extremely ridiculous that this is not a 4 credit course.
	Nerd teaching nerds, Sherriff rocked.
	Sherriff has a great personality, and truly makes class fun and engaging. That being said, I can't say I really learned anything during the lectures, but was rather introduced to concepts to which I would have to go back and learn later. I think his lecture style would be extremely effective if one already knew the material, or had learned it, and used the class as a review. I just think the lectures were so fast that it is hard to learn the material. That being said, I don't know if this is due to Sherriff or just the topic of CS. Regardless, Sherriff made the class fun, which is a heck of a lot better than many other teachers.
	Professor Sherriff is fantastic lecturer and an just an awesome person overall. If I were a little better at math and spatial reasoning and logic I would definitely have switched to this major. Great, great class.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Professor Sherriff is very good at getting people excited about computer science. This class solidified my choice to major in computer engineering or computer science, because I really enjoyed it.
	N/A
	I genuinely enjoyed going to Professor Sherriff's lectures. He not only presented the information in an effective manner, it was entertaining and he didn't always just lecture, he let us interact. I wanted to go to lecture because I knew it wasn't going to be dry and boring and that I would actually learn something relevant to the course.
	I LOVE SHERIFF!!!!
	In class coding is hard to do since we just learned the material. It would be better to go through step by step in front of the whole class rather than let us do it ourselves. Half of the time we had no clue where to start and was difficult. Other than that, the class was fair and enjoyable.
	I hate computer science, but Sherriff is an excellent teacher that understands that computer science does not come easily to everyone.
	The hands on programming in class wasn't very effective for me - I was generally confused on how to actually do most of the coding until after the labs for the respective subjects. The podcasts seemed like they'd be helpful, but were really hard to put together with the class code (which wasn't always easy to find) so it's hard to glean much information from the podcasts. Also, the collab has information all over the place so it's hard to put together all the resources needed to study for tests or do homework. Finally, the slides from this class were absolutely useless for studying and working on assignments. That being said, I loved this class and learned a lot from it - hope the criticism helps make it better!
	I had literally no programming experience prior to this course, and was concerned about its difficulty. I am doing very well in the course, largely in part because of Professor Sherriff's commitment to his students and his course. This was easily the best class I have taken at UVa.
	Excellent course overall, but the final was handled poorly. Ambiguous questions led to confusion for many.
	Awesome!
	I really enjoyed this course in the beginning, but it felt like as soon as we reached loops, things started to speed up significantly and I struggled to keep up with the subject matter. I know that it is unlikely that you would be able to slow down the subject matter, because there is a lot to get through, but in the beginning everything was pretty slow and I understood fairly well, but later on in the course I felt like I was lost. One recommendation I would have is that a lot of time it seemed like we would start doing a homework and have no understanding of what we were suppose to be doing and then after the homework was due, we would practice the material that was on the previous homework. I would personally prefer if this was reversed.
	Compared to what I knew at the beginning of the semester, I have learned a ton.
	One of the best courses I have taken at UVA. Professor Sherriff is a great teacher. I had no programming experience prior to his class, and though I had trouble from time to time. He did a great job teaching the class. Because CS 1110 was so interesting, I taking CS 2110.
	I feel that lecture was sometimes hard to follow, especially if you've never programmed before. I wasn't expecting the class to move so fast in the very beginning, which also made it hard to catch up later
	THis class was good
	Loved the course, it made me change my major to CS. Mark Sherriff is a great teacher, he is really involved in the class and easy to talk to. Even when I some trouble with submitting a homework assignment he was very understanding and pragmatic with how he handled the matter. Go Sherriff! I laughed when filling out number 19 on this form because there is no bubble for "I never got around to buying a textbook." Not saying you are a BAMF or anything... but you kind-of are....!
	I found this course to be very worthwhile. It feels good to have a basic understanding of programming. Still, I sometimes felt that the lectures were much less useful than in other courses. I gained much more from homework assignments and figuring out the problems on my own.
	Great introduction to computer science, Sherriff makes it funny and interesting to non-technical folks
	In hindsight, I should have taken CS 1112 due my technological ineptness. To me, Sherriff had the tendency to breeze over material that I wish he had spent more time reviewing. Perhaps if he spent more time going over examples and having us code with him (at a suitable pace), I would have felt more competent in class. Other than that, I think he did an excellent job instructing.
	Sherriff is probably my favorite professor I've had so far at UVA. He consistently went above and beyond with helping students and with coming up with creative ways to teach programming. I'd say his teaching is a part of the reason I decided to switch my major to CS.
	While upbeat and open during lectures, one on one Mark is unprofessional and patronizing to students. His nasty attitude discouraged me from coming to him for help out of fear that I would be mocked and humiliated for asking stupid questions.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I'm glad it's over.
	Professor Sherriff always keeps the lectures engaging and insightful. TA grading is not fair much of the time. Regrade requests take forever to come through.
	Awesome class and awesome professor!
	good and interesting TAs are helpful overall
	This course is great! I wouldn't say it made me want to major in computer science but I definitely respect the material more! Professor Sherriff is one of the most engaging professors I've had at UVa and is just overall a wonderful professor!
	Good class, I think most of what I learned was from doing the homeworks.
	Sherriff is the man.
	Professor Sherriff was the best professor I had all year.
	I really enjoyed this class. At times it was frustrating because programming is something I knew nothing about before, but I would definitely recommend the course and the instructor. Sherriff, at times, was slightly confusing, but overall, it's obvious that he loves teaching and cares if his students learn the material.
	This course most certainly took me from being utterly uncomfortable with programming and having no experience to feeling like a person with at least some knowledge of programming and what it entails.
	Sherriff is the man
	Great class. Starting off with basics, it wasn't too hard of a class nor did it require a lot of time. I liked that it was helpful and applied to students of all majors, not just engineers. I also enjoyed the homeworks and how we had 2 weeks to do it. Sherriff was a great professor. Absolutely nothing wrong with the teaching. Tests were fair, as well. I have only one negative: the class started extremely extremely easy (like 1+1 stuff). In a matter of a month, and out of no where, it went from easy stuff to extremely "WTF-AHH" stuff. And this is coming from someone pretty good at CS. It wasn't that smooth of a transition. Other than that, I enjoyed this class. I never skipped, no matter how easy the class was. Simply, because Sherriff is awesome and makes this class so bearable.
	n/a
	This course is definitely a course I think everyone should take if they can. Sherriff is an amazing teacher, along with being hilarious at times.
	I had no experience at all with Computer Science prior to this course so I thought that it would be overwhelming but it was not. I feel as though I learned a lot from this course and I enjoyed it very much.
	I felt like the first part of the course went a little slowly and the last part of the course (around classes and methods and onward) could have used a little more development. I took a CS course in high school where we spent more time on graphics, which I think would have been helpful in CS 1110 so that we could actually understand how paint component, Graphics 2D, and other graphics things worked when we used them in the fractals and zombie game. I would replace some of the python coding with graphics lessons, because we never really used python enough for me to truly learn and remember what was taught in the lectures about python and some advanced I/O stuff, whereas we use graphics in later homework assignments so we would definitely learn it better. I'm not sure if partner homework assignments were worthwhile for me or not. Given the difficulty, working in partners probably reduced the amount of work and the number of people asking for help in TA office hours. However, I think I had a more solid understanding of CS by doing the first 3 HWs by myself, and in HW 4 where I was a bit stronger at CS than my partner and felt like I was leading the project. Overall great course, definitely has shown me the usefulness of CS in many different applications.
	Even though I will not pursue computer science further, it was a great class, well organized and Sheriff made the lectures interesting. He was a wonderful professor
	I did not find the tests a good measure of ability. A programming exam should be logic based and not largely on memorization which I found they were, as all programming languages stem from basic logic.
	This was an excellent course! Mad props to Sherriff and his TAs. My only complaint is that Sherriff isn't teaching CS 2110 in the fall.
	Professor Sheriff was one of the best professors I have ever had. He went out of his way to help us understand the material, was funny, interesting, and captivating to listen to, clearly had a complete understanding of the subject, and incorporated creative elements such as chases, technological applications, demonstrations, and activities into the teaching of the material. Overall, he clearly knows what he is doing as I not only learned a great deal from going to class but also enjoyed going. I know a lot of people skipped however because of the podcasts but I feel this benefits everyone because the people who go got more attention and the ones who skipped had free time and still could learn the material while those who were sick or could not make it could still learn the material. In general, I support the use of podcasts even though I never listened to them.
	This class made me want to major in CS

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Such a fun and interesting class, and Sherriff was awesome! I had never programmed before and was a bit worried about the class, but it turned out that my lack of experience was never a concern! I realized writing a code that does what you want it to do is one of the greatest feelings of accomplishments ever... Basically, it was pretty fabulous overall.
	fun course.
	This was a very informative course and gave a great introduction to CS. Unfortunately, I am not a big fan of CS so I didn't love it but Mark Sherriff made it as fun as possible. He is a good instructor and this is a good course.
	This was a very enjoyable course and Mark Sherriff is a great professor.
	The course should be worth 4 credits with the lab credit. There is more than enough work done outside class for this to be a reasonable request.
	Difficult yet rewarding
	Good course.. Sheriff is funny.. grading policy on homework is kinda tough
	GOOD
	The course was taught in just a very poor manner. It is not professor Sherriff's fault at all, in fact I found his lectures significantly more interesting then any other lectures I've had since I've been at UVA. This being said the way that computer science is taught here is just inefficient. Putting code up on the screen is just a poor way to learn. I agree that homework should be a big part of the grade but even so there were many times that I would go into the homework with no idea what to do and I would have to teach myself all of it. I am not sure of a fix for this but I think that there are ways to keep people more involved in the lectures and not on Facebook, because just putting code up on the screen and saying memorize this just doesn't work.
	I enjoyed the course much more than I expected. Sherriff is a great instructor and made classes very easy to pay attention to.
	The homework was too hard compared to what was taught in class. People who had prior programming experience had a huge advantage over those of us who didn't. They thought the homework was generally pretty easy while the rest of us struggled to get through it. Other than the homework which I feel we weren't adequately prepared for, it was a good class with a great teacher.
	Thank you for helping me solidify my decision to major in Computer Science.
	Should not be an engineering requirement.
	I found that the in-class activities were not useful. It felt like Sherriff only had 30 minutes of content and wanted to kill the rest with group work. This would have been more appropriate as ungraded/between-class homework with more focus on the material in class. Also, these moments were super awkward because non-first year/non-SEAS students can't bear working with/have no friends among the first-year SEAS kids. I bought the old textbook from a friend a couple of semesters ago, and I found it a lot easier to use and understand (though it does follow a different progression). I thought this semester's textbook was overly wordy and difficult to use for reference.
	Sherriff should teach more courses because he is a great teacher.
	The only real complaint I had about this class was the tests. During the first test i was sitting on an end chair and two TAs just stared at me while I took the test. I felt so uncomfortable and the TA just kept staring at me. I understand they are there for my own good so that people dont cheat, but I feel I would have done much better without there always being like 10 TAs just staring at all of us. This is a community of trust and I always felt like we werent being trusted at all.
	I had no interest in computer science coming into this course. Now I want to major in computer science. Hands down, this has been my favorite class since coming the university. Professor Sherriff is awesome and clearly loves what he does. His enthusiasm for the subject is contagious.
	Great class, interesting prof. who looks like sheldon cooper from big bang theory
	In class activities were useful. Additionally, I felt that I learned the most in the labs. Doing a small, manageable task that addressed a new concept, with the help of TAs in the lab, was extremely important to my learning experience.
	Professor Sherriff was probably my favorite professor this year!
	Sherriff was the best part of this course. When I was told to consider nominating a professor for the Hartfield-Jefferson Scholars Teaching Prize, my first thought was Sherriff so I wasn't surprised to learn he had already won in 2010.
	I think this is the best run class I have taken here. Also I think I have taken more away from this class, in that I feel very secure with the course material.
	This was one of my favorite courses this semestre. It was really engaging and interesting. Professor Sheriff is a great lecturer, and definitely knows how to explain complicated concepts to large group of people.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	He may have moved a little bit too quickly during lectures for me to understand completely what we were learning every day.
	I thought that class required a lot more work than the 3 credits it counts for. It should be a four credit class. I easily spent over 7-9 hours a week on it outside of the class lectures. Professor Sheriff was excellent and lecturing, but once we got to Arrays and beyond the quality of the lectures decreased. They became a lot more confusing, and he made a lot of assumptions about what the class knew. I think it's important to keep in mind the majority of the class has NO CS experience at all especially when getting into the more complicated topics (classes, methods, etc).
	It would be better possibly to not move so fast during the lessons because some students that didnt have coding experience just felt left behind.
	Sherrif is great.
	Great course. Sherriff makes the class enjoyable.
	I appreciate Professor Sherriff's attempts at making the course more interesting and I really gained a better appreciation for computer science. I did not really like the chases though.
	I felt like I was at a disadvantage from the beginning because I had never taken any type of programming course before this. Because this is an "intro" course, I think it would be more helpful to students like me to maybe move a bit slower, and not to assume we have ANY knowledge of computer science.
	There was almost no way to show which partner did all or none of the work for homework assignments. On one of my homework assignments my partner submitted really bad code very late and I didn't have enough time to rewrite everything. If both partners had to submit the part of the code they were responsible for, the people grading might be able to see how much work each person put in and grade accordingly.
	I think it would be very beneficial to have more than 2 TAs on nights (after 5pm) a week before the homeworks are due. It took sometimes up to 2 hours just to talk to 1 TA, which made it very difficult to make progress on the assignment.
	I really enjoyed the course. I picked mechanical engineering as my major, but if I have time in my schedule, I will definitely take more CS classes. Of all the lectures I've taken so far, Professor Sherriff's were the most engaging.
	Tough course, and it's lot of work for only 3 credit hours (3 hours of class, plus an hour long lab in addition to weekly out of class quizzes and very difficult weekly/biweekly homework assignments). I know this is not Professor Sherriff's fault. Anyway, Sherriff is a great teacher. Makes the lectures fun and interesting. Probably the most enjoyable lecturer I've experienced thus far at UVA.
	This class was awesome. Sherriff is awesome. CS is awesome.
	Prof. Sherriff was an interesting lecturer who made the material easy to understand and exciting.
	Sherriff is one of the best professors I've had at UVa.
	I really liked this course, and enjoyed learning about CS. I know appreciate computer science, and it feels great to know how to program.
	Grading of homework is really slow and some TA are very harsh with grading of HW while some are not based on my observation
	Sherriff is awesome.
	Fantastic Course.
	Good class.
	Awesome class. Learned lots not just about computers but about my own logic and thinking. I think it should be a requirement that everyone at UVA take the class and it should be taught by Sherriff.
	I think this class should be worth 4 credits.
	Some of the homeworks were way too time consuming, also partners did not always work out.
	Moar Cheerwine. And ferromagnetism.
	Professor Sherriff's laid back teaching style was awesome, and he was very understanding and flexible with me when I got the flu and had to miss a test.
	Great course. I enjoyed it very much. I think the pace may have been a bit slow for my liking, but it made the material easy to manage.
	Professor Sherriff is an amazing professor. I looked forward to going to class every day. I don't think I've had a class at UVA that has been quite as amazing as this one.
	I really enjoyed the course--I wish Sherriff would do a 2000-level course next semester!

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	informational class, i didnt really understand some of the material but found that the labs were very helpful
	Prof. Sherriff , you are an awesome teacher. The only negative aspects on this eval are because CS was a class i have been dreading since i got into the E-school. You got me interested in the course a bit, but I dislike everything about it, and it just isnt my thing. But i dont want that to take away from the fact that you are an awesome teacher, and even though i am getting poor grades in your class, I would fail if you werent the teacher!
	I loved it
	Interesting course that was taught very well, stress that keeping up and reading before lectures helps in the long run.
	Sherriff is one of the best, most engaging, effective, entertaining professors I've ever had. A+, sir. He deserves a greater salary than he's got for sure.
	Given the number of students taking this class, it was run extremely efficiently. This class is a great example of how technology can make a large scale operation run very smoothly.
	I absoultely hated the material part of Cs and like never want to program again but Sherriff saved the class for me and made lectures enjoyable and a manageable class to attend. I know some people were probably bored bu i think he went through material in class a little fast and sometimes it was hard to keep up in lecture. Awesome teacher and guy overall though.
	The class difficulty greatly increases after the first test
	None.
	Personally, he has been my favorite professor so far. Very into his work and willing to help out anyone.
	I had never taken a computer science class before this semester and gained a lot from it!
	Great course! Great start to my CpE major!
	This was my favorite class that I have taken at UVA so far. The content was interesting and very useful, the lectures were engaging, and the tests were fair. I learned a ton in the class and can't wait to pursue a computer engineering major. Also, Sherriff is awesome!
	Professor Sherriff and this class were great! Prior to taking this course, I had no interest in majoring in Computer Science. I was only taking it so that I could minor in Systems Engineering. As a result of having Professor Sherriff (and thanks to his enthusiastic teaching), I declared my BACS. Furthermore, I enjoyed this course so much, that I am taking CS 2110 with him this summer.
	Sherriff was an excellent teacher although I'm just not into cs that is why i don't enjoy the course as much.
	Its the kind of class I would bring a prospective student to: its engaging, and takes what could VERY easily be a boring subject and averts its path towards complete and utter boringness.