CS 1110-001 **Introduction to Programming** - Fall 2014
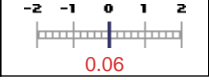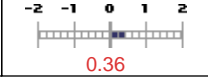
ENGR (17561)

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

Respondents: 81 / Enrollment: 139

Summary: CS 1110-001 Introduction to Programming - Fall 2014 (17561)

Overall Course Rating

CS-1110-001 Mean 3.95
CS-1110-001 Std Dev 1.10
CS-1110-001 Response Count 404

Difference from Category Mean, Expressed in Category Standard Deviations	
	0.06

SEAS, 1000-level courses Mean 3.89
SEAS, 1000-level courses Std Dev 1.04
SEAS, 1000-level courses Response Count 9478

Overall Instructor Rating

INSTRUCTOR: Sherriff, Mark
Mean 4.47
Std Dev 0.71
Response Count 564

Difference from Category Mean, Expressed in Category Standard Deviations	
	0.36

SEAS, 1000-level courses Mean 4.13
SEAS, 1000-level courses Std Dev 0.97
SEAS, 1000-level courses Response Count 14634

~ 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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.23	1.50	24 (29.63%)	16 (19.75%)	10 (12.35%)	17 (20.99%)	14 (17.28%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.23	1.50	24 (29.63%)	16 (19.75%)	10 (12.35%)	17 (20.99%)	14 (17.28%)

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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.35	0.92	44 (54.32%)	28 (34.57%)	4 (4.94%)	3 (3.70%)	2 (2.47%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.35	0.92	44 (54.32%)	28 (34.57%)	4 (4.94%)	3 (3.70%)	2 (2.47%)

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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.60	0.65	55 (67.90%)	21 (25.93%)	4 (4.94%)	1 (1.23%)	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)
81	4.60	0.65	55 (67.90%)	21 (25.93%)	4 (4.94%)	1 (1.23%)	0 (0.00%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

4. How accurate is this statement for you: Pair Programming helped me learn the material better.
~
Question Type: Likert
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.51	1.07	14 (17.28%)	31 (38.27%)	22 (27.16%)	10 (12.35%)	4 (4.94%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	3.51	1.07	14 (17.28%)	31 (38.27%)	22 (27.16%)	10 (12.35%)	4 (4.94%)

5. Which topic/lecture in this course was your favorite and why?
~
Question Type: Short Answer
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Individual Answers
74	*See below for Individual Results*

I found encryption lecture really cool

if statements and loops. They were easy.

I don't know which part was my favorite because everything kind of built upon what we had already learned. That being said, I would say that it was really cool to be able to write classes on our own and have them interact with each other by the end of this course because we used a lot of what we had learned in class to do that.

I liked the intro with the paper airplane to grasp the concept/

I really enjoyed seeing how classes and methods could be created and then tied together and then used in Java!   I also enjoyed game design, although the code was difficult.

I thought recursion was very interesting, yet hard to grasp.

The zombie project really helped me understand the interaction between classes and methods.

Zombies - helped me understand how classes interact and it was fun/rewarding

fractals and algorithms because I enjoyed the problems solving that could be done with cs

Recursion, it was interesting how calling upon the method itself could progress forward.

Recursion

loops because it provided the basis for the more complex code to come

CS Scavenger Hunt was fun.

Class interactions

Zombie game project. Interesting application of programming skills and ideas that make a working final product.

My favorite topic in this course was learning how the "SimpleCanvas", "ZombieSurvival", and the "Human" and "Zombie" classes work together, thereby creating graphics.

I liked recursion because we have always seen recursive sequences in previous math courses, but never created them.

For loops were my favorite. They were fairly simple but they looping through code improved my logic.

TURTLE BOB. I love how we started out with Turtle bob and had no clue what we were writing, but ended with Turtle bob and could understand and appreciate all the methods we wrote.

I liked the fractals! It was fun to program a plant fractal, because I had to go in and fix the turtle code.

ArrayList, Loops because they were the most useful

recursion

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Writing methods, because it helped me to understand how programs can run more smoothly and how to make classes interact with one another, giving me more flexibility in things I would want to do in a program.

Writing methods, because it helped me to understand how programs can run more smoothly and how to make classes interact with one another, giving me more flexibility in things I would want to do in a program.

I liked turtle drawing because it made computer science feel fun and manageable for me. I think I learned the most from doing the Zombie project.

The last lecture were we could ask whatever questions about computers we wanted to. I actually learned a lot that day.

Favorite topic was methods because they are interesting

My favorite was learning how to write methods, since programming the games were a lot of fun!

I enjoyed loops and arrays/arraylists because it was very straightforward and I thought it was interesting.

I really enjoyed building the Zombie game, it allowed us to use everything we had learned about classes and really make something for ourselves.

Encryption was probably the most fun because we got to go on a scavenger hunt but also because cracking codes is interesting.

The lecture on ciphers because the chase was fun and interactive

methods because i found it the easiest to understand

The idea of methods inside classes, classes inside classes was super awesome. I also loved how I could use a code written by another person so easily.

Methods and Classes. Really showed how real programs are built rather than programs just written in the main method.

My favorite topic was recursion because it was interesting to come up with the solutions.

My favorite topics were the turtle drawings and the lecture with the Frisbee, which explained how things are stored and called in a program.

Email hunt was fun

I liked creating object classes and applying them, such as with the POTDs for the schedule, but the simple word-games like Nim were very cool too. I liked that they felt applicable to real life. I could use the schedule to help me build my own schedule. It felt like a step forward to building the kind of apps I'd be most interested in.

Zombie because it helped me learn various topics such as ArrayList that I didn't exactly understand wel before.

Recursion was an interesting topic and something that I felt is applicable in other areas.

I really enjoyed when method and class interaction finally clicked. It was at this time that everything else in the course fell together and made sense for me so I could really understand what was going on.

I liked for loop because it was the topic that was most interesting to me.

Creating classes because it tied everything together.

I really enjoyed encryption and recursion. I really enjoyed the more conceptually challenging material, because it makes so much logical sense, but it was challenging to actually implement.

Recursion and fractals were my favorite. It was the most difficult topic for me to master, but the logic behind them was very cool. It made me quite happy.

Recursion by the end of it, I like the logic

Turtle because it's fun and easy

The game project because it helped me understand the basics of getting objects on a screen to interact with each other.

Classes and methods, it tied it all together

I think the scavenger hunt was my favorite because it was a fun activity that allowed us to apply the class material to a fun game.

My favorite topic in this course was recursion. While challenging at first, it was extremely interesting to explore the different things we could do with recursion.

Loops. Just a cool concept.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Recursion, it was an interesting way to solve problems.
	I didn't have one specific topic I like more than others. Mostly, I liked learning how to code and stepping through the logic.
	Decoding scavenger hunt. It was cool to get out of the classroom and the challenge was fun.
	I liked learning about encryption because it seemed the most mysterious and interesting to me.
	I really liked when we put it all together in the final POTDs and wrote our own classes and understood how objects can really work together.
	Loops because they are most useful.
	Recursion - it is an interesting new way of thinking about a problem
	Loops
	Building classes because it allowed for freedom of design
	Data types because it was easier to understand and fundamentals are important
	GUI
	Early lectures with the lego analogies - made it easy to understand what part of a programming language are
	Learning to program games. I have always loved games, and learning how to program them helped me to appreciate their production.
	Writing classes was my favorite part of this course because it expanded the boundaries of computer science and its possibilities for me, personally.
	I enjoyed learning about methods because it became easier to write programs.
	cyphers
	I liked the exercise involving frisbees where Sherriff showed us how memory storage works.
	Professor Sherriff addressed a number of great topics. My top 5 favorites were as follows: 1. The Human Recursive Machine- I would never have understood recursion without this lecture. It was a terrific analogy to the process of recursion within a computer. 2. The Paper Airplane Algorithm Lecture- Another terrific physical analogy to the process which the computer undergoes. 3. The Rapid Fire Topics Lecture- It was truly inspiring to visit so many topics in under an hour. It definitely made me more interested in the field of computer science and was extremely entertaining. 4. The Case for Computer Science- Professor Sherriff immediately dispelled my initial apprehensiveness about where computer science would apply in my engineering career. 5. The Final Exam Review- Prof. Sherriff did a great job preparing us for the exam. His final speech was indicative of what an excellent and caring professor he is.
	Class, it was a starting point to build a structured program
	Hello World - because it was just the beginning
	Methods. I liked the way they worked and just got a very specific task done.
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-1110-001, Sherriff, Mark	
Total	Individual Answers
73	*See below for Individual Results*

	I think I will find the debugging we did (not as much during lectures, but in POTD's and stuff) the most useful
	Probably none, given this is the only programming class I'll take. Maybe loops.
	Class Interaction,
	everything
	basic coding logic that can be translated to other fields
	The ability to problem solve and have strategies to problem solve.

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I think method and class interaction will be the most useful because it makes you think in a different way than you're used to, at least for me. It allows you to take different views on problem solving, and not just in CS.
	Recursion
	understanding how to think in a way that works with computers will help me later in life
	just learning to program in general
	How to break down a problem into the simplest steps and be able to communicate that.
	The scope unit was most helpful to me. Explaining how to write methods and classes and how to make them interact will be extremely helpful to me in the future.
	Learning the logic-based approach at solving problems.
	Definitely creating object classes.
	Methods, Classes, and recursion.
	recursion, how to write classes and methods
	none
	algorithms
	algorithms
	recursion
	Every lecture with an analogy
	Scanner method
	creating classes and methods.
	data types, loops
	if else statements
	The demonstration of how the different methods/stacks.
	How computers work and how applicable problem solving is to other programs.
	ArrayList, Loops, writing methods
	Not really a topic, but CS is really all about problem solving so now I have a lot of practice solving all different kinds of problems in all different kinds of ways.
	Recursion will be very useful to me because it required the most thought and a higher ability to look into problems
	N/A
	I think I might use the skills I developed in understanding how programming decision structures and loops works in forming algorithms.
	Basic understanding of computers and coding
	For some of the labs and homework assignments, we had to use the code written by the professors. I think this is a super important skill, as I can use codes written by others (of course they should be the ones shared public) and enhance my own work.
	I don't know.
	Just the basic ideas of coding.
	talking between classes i think would be the most useful for the future
	The overall understanding of coding and the fundamental concepts of how a programming language works.
	GUI, RECURSION
	Being able to write methods and have classes read from each other is the most useful
	Just learning how to write basic java code i.e. while loops, for loops, if/else statements gives an interesting perspective on how to approach problems.

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The way of thinking about algorithms that it teaches will be very helpful.
	For loop
	No single lecture. Learning the logic pathways and gaining a proper understanding for computer science as a whole was the most useful part of the course.
	The syntax of the language.
	Ways computing can be applied lecture or the programming languages lecture
	I think the fundamentals such as writing classes and methods will be most useful in the future as these concepts can be applied to other programming languages.
	The end of the course speech - cs is everywhere and I learned about the art of cs and the skill of programming...
	Learning how to think computationally (creating an algorithm) to solve a problem
	Iteration
	Learning how to design and build functioning multi class programs
	I feel like learning the basics of Java will lend itself in helping us learn other computer programming languages.
	constructing classes
	loops
	writing loops
	Difficult to pick just one.
	All of it. Just the problem solving mindset you need in CS is what will be most useful in the future.
	Classes and algorithm design because understanding how classes work is extremely important and algorithm design can make a program better.
	Honestly, none of the lectures is extremely helpful on its own. However, the Arrays and ArrayLists Lectures will probably be the most valuable going forward.
	Learning basic coding. All of it, hard to pick one. The lectures were broad and taught basic coding.
	Class, it will help me build a program
	The problem solving understanding that it emphasizes will be extremely helpful for my approach to problems in the future.
	I will probably find the topic about loops and scope helpful in the future.
	The email hunt project will probably come in handy when I write an internet scam that makes me a lot of money
	Being able to read Java
	Array Lists
	ArrayList, arrays
	Writing methods and figuring out how to trace errors.
	Just programming in general is a great skill to have in today's job market.
	Algorithmic thinking
	Logic based coding questions really challenged me to think in efficient ways about problems. Emphasis on different ways of solving problems, through incremental counting and sorting.
	The ability to solve problems using a program will definitely help in the future
	Basic understanding of different loops, because it will transfer into other languages.

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

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-1110-001, Sherriff, Mark	
Total	Individual Answers
67	*See below for Individual Results*

Drawing pictures

Honestly, every lecture was helpful in a different way. I was completely clueless about CS at the beginning of the semester, but Prof. Sherriff's lectures brought me up to speed such that I was able to achieve an A even after receiving a 75 on the first test.

decryptions

Arrays

Recursion

I think that learning different types of algorithms (sorting and searching) were useful, but we didn't use them much in assignments.

History of coding.

I thought most lectures in this class were somewhat useful.

UML diagrams took me a while to understand, but now I think they will be beneficial for designing solutions to problems. Cookie clicker didn't really do it for me. Kinda pointless way to teach mouseAction.

None that I can think of.

recursion

After thinking that iterative programming methods were taught clearly, I think that the recursion lectures were less clear

Classes were hard to understand. I got really confused by null and full constructors in the lecture, but got it later, once we started using them a lot.

Recursive method

Nothing really. It all seemed applicable down the road.

I had trouble with the graphic games (zombies, for example). I have trouble understanding what exactly is happening (since many of the classes are already given to us, there seems to be a lot happening behind closed curtains). For example, I don't fully comprehend what Graphics2D g is, or how I can implement it myself (it's already given to us in most/all cases). Also, unless you'll become a games-programmer, I didn't feel that the knowledge behind these games was particularly useful or applicable to other programming areas.

drawing fractals

Learning all the parts of the physical computer seemed irrelevant.

The class on different searches and sorts did not work for me. I was having trouble grasping the complexities of each type.

recursions, csv

Turtles

JOptionPane was the only topic that was not useful in the long term.

The last few lectures about more theoretical topics

I don't think anything applies here. Most topics at first were confusing because they were new, but once I understood them, they were all useful because the course continues to build off of prior information.

all were ok.

I wish we had more chance to work with JOption or learn how to use graphics more.

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The detailed discussions about how computers work was not very applicable to the rest of the material. Also having a good programming style was probably not at important at all for students like me who don't plan to major in CS.
	I did not find it helpful to learn about the inner-workings of the computer, the computer engineering topics did not interest me nearly as much as the computer science topics.
	I'm not sure. I generally found everything we learned to be pertinent.
	The sorting complexities doesn't seem useful to me in the long run.
	I think that we could have spent more time on bubble sort/merge sort
	recursion, recursion, recursion
	I didn't really get a lot out of learning recursive methods. I could see where they might be useful, but iterative methods were just so much easier to understand and use to solve real methods.
	Components about physical computers and how they worked lost me and didn't help my overall understanding of computer science.
	arrays, arraylists - examples in class weren't helpful
	Recursion and Hardware information
	I think everything taught was valuable.
	Probably lots of the stuff we did with turtle in the beginning of the year, but I found just about everything we did to be extremely useful in the future.
	Recursion is cool and pretty interesting, but I don't really see its function given that loops can often be used.
	Turtle drawing.
	None
	None
	While recursion may be helpful, it was hard for me to grasp.
	the speed of the course was too fast
	Recursion. It melted my brain and honestly it seemed like most things you did with recursion you could do through other means.
	I did not find recursion as useful, especially since I think loops could work just as well.
	I think the search and sort methods were pretty useless unless you intend to delve deeper into computer science. They just seemed thrown in there and knowing how they worked didn't apply to the course at all
	Turtle Drawings
	Recursion.
	I didn't really see any topic that were not useful.
	I thought labs were a bit strange. They weren't often related to what we were learning in class. Usually it was just a time of trying to figure out something you hadn't learned.
	Not any, really.
	Encryption chase was fun but maybe not that necessary. Also if you had a partner that understood it you didn't have to understand it and they did it all which made it kind of pointless for one person.
	n/a
	n/a
	nothing really.
	the lecture on different types of programs as well as the rapid fire although it was interesting, i feel it would've been more productive to do exam review
	Some of the labs didn't relate well to the topics we were covering in lecture and I felt that some of them were pointless.

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

the lectures were only useful about 30% of the time. most of the time, i left lecture feeling like i wasted my time and like it wasn't going to help me on the homework assignments. i learned everything from the textbook, but that didn't cover everything in class.

Recursion through drawing was intuitive at first, but we also need to learn (in class) how it's applied in other code.

Recursion, they are not required, and could be handled much easier with other loops

Maybe the turtle part.

The least useful topic for me was the game project. The concepts were interesting, but in my career path, the specific logic paths won't be very useful at all.

garbage collection

Rapid Fire Topics

---

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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
79	3.88	1.11	20 (25.32%)	13 (16.46%)	14 (17.72%)	3 (3.80%)	2 (2.53%)	27 (34.18%)

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)
79	3.88	1.11	20 (25.32%)	13 (16.46%)	14 (17.72%)	3 (3.80%)	2 (2.53%)	27 (34.18%)

---

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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
79	3.76	1.10	17 (21.52%)	17 (21.52%)	14 (17.72%)	5 (6.33%)	2 (2.53%)	24 (30.38%)

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)
79	3.76	1.10	17 (21.52%)	17 (21.52%)	14 (17.72%)	5 (6.33%)	2 (2.53%)	24 (30.38%)

---

10. How often did you listen to the podcast for a lecture?
~
Question Type: Multiple Choice
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Every lecture (NA)	Nearly every lecture (NA)	Whenever I needed to review a topic (NA)	Only when I missed a class (NA)	Randomly just to see what it was like (NA)	Never (NA)
81	1 (1.23%)	1 (1.23%)	20 (24.69%)	17 (20.99%)	11 (13.58%)	31 (38.27%)

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)
81	1 (1.23%)	1 (1.23%)	20 (24.69%)	17 (20.99%)	11 (13.58%)	31 (38.27%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

11. How would you rate the availability of TAs?
~
Question Type: Likert
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
81	2.78	0.95	18 (22.22%)	37 (45.68%)	17 (20.99%)	8 (9.88%)	1 (1.23%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
81	2.78	0.95	18 (22.22%)	37 (45.68%)	17 (20.99%)	8 (9.88%)	1 (1.23%)

12. How would you rate the helpfulness of the TAs?
~
Question Type: Likert
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
80	3.02	0.84	23 (28.75%)	41 (51.25%)	12 (15.00%)	3 (3.75%)	1 (1.25%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
80	3.02	0.84	23 (28.75%)	41 (51.25%)	12 (15.00%)	3 (3.75%)	1 (1.25%)

13. How often did you make use of the TA office hours?
~
Question Type: Multiple Choice
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Every week (NA)	Every other week (NA)	Once per assignment (NA)	Rarely (NA)	Never (NA)
81	9 (11.11%)	13 (16.05%)	13 (16.05%)	34 (41.98%)	12 (14.81%)

Results for SEAS, 1000-level courses

Total	Every week (NA)	Every other week (NA)	Once per assignment (NA)	Rarely (NA)	Never (NA)
81	9 (11.11%)	13 (16.05%)	13 (16.05%)	34 (41.98%)	12 (14.81%)

14. Any specific comments about the TAs you would like to share?
~
Question Type: Short Answer
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark

Total	Individual Answers
47	*See below for Individual Results*

Really liked them, when I had a problem, they were there for me and did not just give me code, but helped me understand conceptually and made me understand and write the code.

A lot of the TAs were awesome. Some of them had no idea what was going on. I think it would be awesome for them to either have to code the POTDs before coming to help or be able to look at some kind of answer code so they at least have an idea of how to help students/point them in the right direction rather than having to themselves work through all the logic and get as far as the student looking for help and get stuck in the same place. this was frustrating at times.

I wouldn't have survived without them. They are fundamental to the course.

Sometimes really helpful, but often times not at all. I would ask them a question and they would then answer it in a way that I usually didn't even understand and then run off before I could get a clarification.

I understand that the TA's were busy and frustrated, especially with so many students. However, I wish they were more patient and willing to explain the problem instead of just point at a part and saying that's where the problem is. This isn't to say they were all bad, there were a few good TAs

They seem like a really fun, friendly group of people. That's part of the reason I want to be a TA.

More TAs per lab session would be better. There were times the TAs were busy helping other students, which was a little inefficient.

I think it is imperative for people to come to office hours with specific questions. A lot of times, kids higher up in the queue would ask questions like "what's wrong with my method" and to figure out the answer, a lot of time is taken since everyone writes methods in their own way. I thought that once I was able to speak to the TA's, they were phenomenal and had a good grasp of the material

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	They all did a great job!
	The only interaction I had with the TAs was in lab, where I found them very approachable and helpful.
	Too swamped during peak office hours
	The TA's were good and helpful, but lab for this class was absolutely useless.
	Graham is great
	There was one named scott that was great. I like scott.
	It was a hit or miss. Some of the TAs were very helpful and great catalysts for learning. Many of the TAs sucked though, and didn't know what they were talking about or were apathetic. Not enough in general because so many kids go to office hours during major assignments.
	More TA's would have been helpful.
	Scott is awesome
	none
	One particular TA was very condescending.
	None
	None
	They were generally very helpful.
	Make more available in stacks when a project deadline is coming up.
	They were never available for one on one help because there were not enough of them
	Some TAs were very very helpful, others were not there wasnt much consistency
	They were all good, but for the assignments that were especially difficult there seemed to be shortage of TAs because of the excess of students who needed help. So having more TAs would be ideal
	I wish some of the TAs actually did the programs themselves because sometimes they were lost and couldn't help us.
	I found the TAs were very helpful, but I wish there was a better way to get help when the queue is too full.
	I hope they are more supportive in labs, even if they know that they would give full credit even if the submitted works don't work properly
	I didn't meet all of the TAs but 90% of the ones I did were extremely helpful. They were able to spot what was wrong and explain why that wouldn't work and guide towards the thinking that would put you on the right path. They were also really nice and would stay late to help if there were lots of people there before a big assignment was due.
	Some TAs seemed annoyed when you asked too many questions in lab
	N/A
	N/A
	No
	TAs had office hours at all the wrong times and never when the most people needed help
	They were great in lab
	Some TAs were not at all efficient with budgeting time. One day in particular I was at every available office hours for a day (totaling around six hours) and a TA was only able to help me once. Some would spend periods of more than 30 minutes with one student. Justin is absolutely phenomenal. He was always helpful and approachable, and he made sure not only that I understood whatever it was that I was fixing, but also to move from student to student as efficiently as possible.
	TA's were helpful with specific questions, but as soon as I took a few seconds to think about what else I was having trouble with, they would rush and go to someone else. I understand many students need help at office hours, but I often felt rushed and not thoroughly helped.
	Some are better than others.
	The TA's were awesome, but lots of times they were overwhelmed and hurried because there were so many people needing help during lots of the big assignments

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Some were definitely more helpful than others. Thank you for doing what you do! Try to better understand the students question before diving into an answer. Many times I had a TA answer a question that I never asked. Not really Way too confusing to figure out where and when they were available. It seemed older TA's were more helpful, while new or 2nd year TA's didn't help much... There need to be more TAs at office hour times, especially for the bigger projects at the end of the year. When the all of stacks are filled with CS students waiting for one TA, the TA cannot spend as much time helping each individual and is spread extremely thin. It would be better if there were at least two TAs circling to help. Also, if the same question is being asked multiple times, it could be beneficial for one TA to explain the solution to that question while the other TA circulates to address other questions. Overall, the TAs were extremely helpful, just too limited in quantity. They were great!

15. What other topics do you wish we had time to cover or which topics did we cover that you wish we could have covered more deeply?
~
Question Type: Short Answer
~
contributed by Sherriff, Mark (mss2x)

Results for CS-1110-001, Sherriff, Mark	
Total	Individual Answers
53	*See below for Individual Results*

Although methods are very important in CS, I did not learn as much about them and I wish we went over them more.

The class did not really teach game programming very extensively, but expected us to understand how to do it in assignments. I wish we had gone over this in the Lectures.

things covered in labs (making games)

N/a

Graphics design/ creating worlds

I wish we did more projects

Recursion. Also GUIs.

Recursive coding. The format for certain codes.

algorithms

I wished we had done some things with hacking. That would have been super cool.

Would've been cool to do the GUI at the end of the year with loading in pictures, like on a green screen. I would have preferred that over the rapid-fire questions we did.

I wish we spent more time on methods and classes, I feel we rushed through them and I wasn't prepared for the test.

I would have liked to learn about how our knowledge of code can be used to create applications or such.

Nothing.

I wish we could have spent more time writing methods and making classes interact with each other. The scheduled lecture time of 50 mins did not seem enough for me.

Encryption

I felt everything was given a fair amount of class discussion.

N/A

N/A

Maybe more encryption/decryption (but more as a bonus at the end of the course)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Computer security
	For each loops
	Spending more time discussing the differences and pros and cons of the various programming languages would be nice.
	More on concepts of DATA storage and how that applies in the real world. More logic based problems. Eclipse is great for allowing some leeway on syntax, thank you! More examples of cool ways CS appears in different areas of studying around the world. What a real world start-up group programming looks like.
	I think it would be helpful for you to make us code by hand for some assignments. that would suck to grade, but I think it could be good for us. Like on the final I had no idea how to write getters and setters because I never had to.... Not sure exactly how you would do this. Maybe like on test review days have everyone take out a piece of paper and do a coding question by hand? Or maybe its really just the student's responsibility to practice on their own (and in this case, you could emphasize this suggestion more often?).
	I liked most of the topics covered and how the course flowed
	I think recursion could've been taught more clearly
	Website design
	A brief intro to another programming language
	I wish we had done more recursion and maybe spent some time doing either a recursion POTD or doing more complicated examples
	More about collisions in game design and what not and how to make them smooth.
	nothing (i really dont know much else about coding)
	Recursion, what a CS major might do in a job
	Langauges or than Java
	Arrays, ArrayLists, Loops, & (how objects are passed to a method) should be covered in depth because they are needed in every assignment as the course progresses. No suggestions on additional topics.
	None
	None
	None
	I never got a firm grasp on recursion. So if we need to learn it then a more in depth lecture(s) would be helpful
	Game programming
	I wish we had covered GUIs.
	I have nothing to offer here.
	Do while loops
	I would have liked to get more exposure to other languages
	I really enjoyed the rapid-fire topics lecture. Sherriff's substantial insight into every topic that was brought up was very interesting.
	Nope, the course was well designed and comprehensive.
	Class inheritance
	JOptionPane stuff
	Algorithms
	I wish we had gone more deeply into classes working together and recursion
	graphics, functions
	recursion could have been covered more deeply because it is difficult to grasp

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

16. 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-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.33	0.69	35 (43.21%)	40 (49.38%)	4 (4.94%)	2 (2.47%)	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)
1892	4.11	0.88	672 (35.52%)	894 (47.25%)	208 (10.99%)	76 (4.02%)	34 (1.80%)	8 (0.42%)

17. 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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	4.54	0.65	47 (58.75%)	31 (38.75%)	1 (1.25%)	0 (0.00%)	1 (1.25%)	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)
2097	3.99	1.11	780 (37.20%)	649 (30.95%)	261 (12.45%)	143 (6.82%)	85 (4.05%)	179 (8.54%)

18. There was a reasonable level of effort expected for the credit hours received.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	3.90	1.28	34 (41.98%)	27 (33.33%)	4 (4.94%)	10 (12.35%)	6 (7.41%)	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)
1899	4.03	0.98	642 (33.81%)	910 (47.92%)	164 (8.64%)	115 (6.06%)	62 (3.26%)	6 (0.32%)

19. The homework assignments helped me learn the subject matter.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.48	0.59	43 (53.09%)	34 (41.98%)	4 (4.94%)	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)
1896	3.99	1.01	636 (33.54%)	758 (39.98%)	267 (14.08%)	108 (5.70%)	59 (3.11%)	68 (3.59%)

20. The textbook increased my understanding of the material.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	3.32	1.17	16 (19.75%)	15 (18.52%)	30 (37.04%)	10 (12.35%)	6 (7.41%)	4 (4.94%)

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)
1901	3.39	1.17	251 (13.20%)	423 (22.25%)	412 (21.67%)	164 (8.63%)	117 (6.15%)	534 (28.09%)

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

21. The course material was well organized and developed.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	4.31	0.82	38 (47.50%)	33 (41.25%)	6 (7.50%)	2 (2.50%)	1 (1.25%)	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)
2082	3.96	0.99	650 (31.22%)	879 (42.22%)	291 (13.98%)	134 (6.44%)	55 (2.64%)	73 (3.51%)

22. The instructor was knowledgeable about the subject matter.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.74	0.44	60 (74.07%)	21 (25.93%)	0 (0.00%)	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)
2095	4.40	0.79	1068 (50.98%)	670 (31.98%)	178 (8.50%)	32 (1.53%)	16 (0.76%)	131 (6.25%)

23. The instructor was well prepared for class.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.63	0.56	53 (65.43%)	27 (33.33%)	0 (0.00%)	1 (1.23%)	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)
2090	4.28	0.85	906 (43.35%)	694 (33.21%)	201 (9.62%)	59 (2.82%)	18 (0.86%)	212 (10.14%)

24. 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-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	3.42	1.22	9 (11.25%)	11 (13.75%)	11 (13.75%)	6 (7.50%)	3 (3.75%)	40 (50.00%)

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)
1890	3.73	1.05	290 (15.34%)	372 (19.68%)	303 (16.03%)	75 (3.97%)	43 (2.28%)	807 (42.70%)

25. The grading policy was fair.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.22	0.71	29 (35.80%)	43 (53.09%)	7 (8.64%)	2 (2.47%)	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)
2095	3.93	1.05	651 (31.07%)	856 (40.86%)	259 (12.36%)	165 (7.88%)	70 (3.34%)	94 (4.49%)

The information in this document is private and confidential. Please handle accordingly.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

26. The instructor responded adequately to in-class questions.
~
Question Type: Likert
~
contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
80	4.29	0.90	39 (48.75%)	31 (38.75%)	6 (7.50%)	2 (2.50%)	2 (2.50%)	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)
2086	4.16	0.95	797 (38.21%)	766 (36.72%)	192 (9.20%)	84 (4.03%)	46 (2.21%)	201 (9.64%)

27. 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-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
81	4.58	0.63	52 (64.20%)	25 (30.86%)	3 (3.70%)	1 (1.23%)	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)
2089	4.18	0.88	820 (39.25%)	773 (37.00%)	260 (12.45%)	65 (3.11%)	26 (1.24%)	145 (6.94%)

28. The average number of hours per week I spent outside of class preparing for this course was:
~
Question Type: Multiple Choice
~
contributed by Office of the Provost

Results for CS-1110-001

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
80	0 (0.00%)	20 (25.00%)	43 (53.75%)	14 (17.50%)	3 (3.75%)

Results for SEAS, 1000-level courses

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
1899	189 (9.95%)	963 (50.71%)	555 (29.23%)	148 (7.79%)	44 (2.32%)

29. I learned a great deal in this course.
~
Question Type: Likert
~
contributed by Office of the Provost

Results for CS-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
80	4.50	0.69	48 (60.00%)	25 (31.25%)	6 (7.50%)	1 (1.25%)	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)
1890	3.90	1.04	588 (31.11%)	796 (42.12%)	306 (16.19%)	125 (6.61%)	75 (3.97%)

30. Overall, this was a worthwhile course.
~
Question Type: Likert
~
contributed by Office of the Provost

Results for CS-1110-001

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.36	0.90	45 (55.56%)	25 (30.86%)	8 (9.88%)	1 (1.23%)	2 (2.47%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
1896	3.86	1.14	645 (34.02%)	713 (37.61%)	284 (14.98%)	143 (7.54%)	111 (5.85%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

31. The course's goals and requirements were defined and adhered to by the instructor.
~
Question Type: Likert
~
contributed by Office of the Provost

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
80	4.42	0.71	41 (51.25%)	34 (42.50%)	4 (5.00%)	0 (0.00%)	1 (1.25%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2089	4.11	0.87	736 (35.23%)	986 (47.20%)	257 (12.30%)	74 (3.54%)	36 (1.72%)

32. The instructor was approachable and made himself/herself available to students outside the classroom.
~
Question Type: Likert
~
contributed by Office of the Provost

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
80	3.69	1.11	21 (26.25%)	28 (35.00%)	20 (25.00%)	7 (8.75%)	4 (5.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2094	4.04	0.97	784 (37.44%)	797 (38.06%)	373 (17.81%)	89 (4.25%)	51 (2.44%)

33. Overall, the instructor was an effective teacher.
~
Question Type: Likert
~
contributed by Office of the Provost

Results for CS-1110-001, Sherriff, Mark

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
81	4.38	0.77	41 (50.62%)	33 (40.74%)	5 (6.17%)	1 (1.23%)	1 (1.23%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
2099	4.01	1.01	781 (37.21%)	796 (37.92%)	348 (16.58%)	110 (5.24%)	64 (3.05%)

34. Please make any overall comments or observations about this course:
~
Question Type: Short Answer
~
contributed by Office of the Provost

Results for CS-1110-001

Total	Individual Answers
41	*See below for Individual Results*

Thanks for a great year!

Lab was a joke. Didn't apply to lessons at all, such a waste of time. And for 3 credits this class was way too much work and class time

Very impressed by this course

One of the best intro classes I have taken thus far.

Phenomenal course! I came into SEAS undecided (though leaning to systems) and have now decided to switch to computer science. For a large lecture course, it was incredibly well organized and after hearing a lot from my friends about their school's intro CS courses, I think this is one of the best in the country. Professor Sherriff presented the material in a very engaging and clear way. Specifically, I think he gave an amazing lecture on what recursion was and how it works and another on about memory/CPU/, etc. Thank you for everything!

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

The only problem I had with the course was Zombies. We never talked about game logic in lecture, and we didn't really discuss it in lab either - we were sort of thrown into the deep end with Flappy Bird. I understand the idea that we might learn as we went with Flappy Bird and then we could apply that knowledge to Zombies. However, as was probably the case for other partnerships, we didn't really have a strong hold on what we were doing in Flappy Bird; the game logic was not intuitive. Therefore when we went to do Zombies, we had to seek a fair amount of explanation from TAs because we simply didn't know how to program a game. Also, the directions were often confusing. Again, I understand that they were in place as more of a guideline than a step-by-step how-to, but with directions on both the website and in the code it was hard to know what exactly was expected. I just think that Zombies counted for a very large chunk of our grade considering how much time we actually spent talking about gaming. Overall I really enjoyed this course. CS is very different from other subjects and it's fascinating. I felt that the homework assignments were an appropriate level of challenge and I was always really proud of myself when I figured them out - that special satisfaction is not a feeling that you get in many other classes!

I think Professor Sherriff could make the lectures a little more structured. He bounced around a lot, talked pretty fast, and the notes he wrote on the board were pretty difficult for me to follow. Also was a bit condescending at times when answering some questions during class. I did really appreciate his enthusiasm in the class though, and he certainly was well versed in the material.

The class was overall great and I highly recommend it to other students interested in computer science. However, one thing I did not enjoy about this class was the partners for projects had to be someone in our lab. My partner was very unreliable and irresponsible in terms of being prepared, communicating, and showed up upwards of 1-2 hours late to every meeting.

I love computer science but I'm going to be a computer engineer.

Great course! Sherriff totally deserves the outstanding reviews I heard before the class! Definitely one of the best professors I've had in my three years here.

Mark Sherriff was the best teacher I've had thus far (not that far considering i'm a first year) but he truly made an impact on my appreciation for computer science and has inspired me to want to do something with it, even if it isn't my major. His teaching style was incredibly accessible and I looked forward to this class every monday wednesday friday. I have so much respect for him as a professor and also as a person. He is so intelligent and really cares about his students which is very apparent. I went to him seeking advice about completely unrelated things (course selection for the spring) and he was more than willing to provide advice and guidance in a kind, straightforward way. Overall an incredible class, and an incredible professor. Very grateful to have taken this class with Sherriff.

Professor Sherriff might be the single greatest professor at this university. If I were his boss, I would give him a fat Christmas bonus...

Mark Sherriff was condescending in his office hours and not very helpful. He does not seem friendly outside of class.

I wish the four our gap for assignment feedbacks can be reduced. Also I hope the design of the curriculum can be more systematic.

Thank you!

Obviously actually reading the textbook is up to the student and no one can make him or her do it, but having read the textbook along with the class (albeit I started after the second test) I can say it was extremely useful for understanding the course material outside of the classroom. I think some people may underestimate the book and some have never even opened it, but I think it should be made clear to the students that it is actually a great resource.

Having never taken a CS course before, I was nervous going into this course. However, I can say for certain that it was one of the best courses I've taken so far in college (as a 2nd year).

A lot of work for 3 credits. Lab 13 and 14 were really tough and a lot of time was put in for 1% and I doubt i got full credit.

MORE TA'S PLEASE!!!

I dread going to Prof. Sherriff's OH because he's kind of mean and I feel like he thinks my questions are stupid. I go to TA's OH instead but it takes forever because there's 1 TA helping 20 or more students.

A great course that has inspired me to pursue CS as a career. Interesting material made even better by a great instructor.

Fun course and the lectures were excellent. It's hard for the TA's to sometimes translate their thoughts into coding help but they made it work. Sherriff was an excellent lecturer making eleven o'clock an always enjoyable time.

Good experience but POTD could be very frustrating and time consuming at times.

Lecture often seemed unnecessary and unfocused on the current topic

No comments.

It was fairly more difficult than I was expecting for an introductory computer science course, mostly towards the second half of the semester.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Sherriff definitely is highly knowledgable in the subject and taught it very well. Anyone who is interested in CS should take his class
	Great professor; wonderful class and I learned a lot. Professor Sherriff presents lectures very well and is very knowledgeable about the subjects. Assignments are structured well so that students learn all the material through doing. As mentioned in last day of class, Sherriff is somewhat unapproachable during office hours and sometimes a jerk. However, at least he realizes this and hopefully will get better at it. The curse of knowledge can be very frustrating, but no need to be mean. Overall, I would say great instructor and well designed course.
	This should be a 4-credit course. Ridiculous that it is not.
	The time expected from students outside of class can be extremely high. The assignments at the beginning of the semester are due every 2-3 days, and getting a basic grasp on the material can be hard, especially when the only non-human resources allowed are lecture notes and the textbook, and you cannot receive coding help from other students. Later in the semester, the first project is not too bad, but the second project is very difficult and time-consuming. The first project uses the same coding application as is taught in the lecture. However, the Lecture taught absolutely nothing about how to apply code to game design, and students had to just find a way to figure it out for the second project. The saving grace here was the high availability of TAs during this time. I was averaging around 7 hours per week outside of class trying to either study, do homework, work on projects, or get TA help with any of these. In the end, though, the course taught me a lot and I found it very worthwhile, though also incredibly expectant of its students for an intro course.
	Great
	I wish there would be less POTDs and projects going on at the same time. Sometimes we would have both POTD, projects, and lab work in the same week and it takes up too much time.
	I would recommend this course to everyone even if they are not interested in CS major or minor.
	NOne
	I really enjoyed the course.
	Instructor was not helpful during office hours.
	Sherriff is the bomb, and his daughter's really cute. It makes me really sad that he doesn't teach 2110 or 2102.
	Great course.
	Professor Sherriff was a very effective teacher who was also able to be entertaining and personable, even in a large class. He also truly enjoys what he is doing.
	I loved this course! Sheriff is a great instructor and makes every class worthwhile. I felt bad if I missed a lecture. The podcasts are excellent too, but it'd be helpful to have the code for the class too (sometimes we had it, sometimes we didn't), because at many times he would say something like "now we do this," and I wouldn't be sure what "this" referred to. Other than that, this was an amazing class, and I'm not sure how you could improve it. Also, Piazza is awesome and everyone is super helpful!
	Easily my favorite class of the semester! Great teaching staff, interesting content, and useful skills learned for my next four years in engineering.