CS 1110-002 **Introduction to Programming** - Spring 2015

ENGR (17208)

INSTRUCTORS: **Tychonievich, Luther (lat7h)**

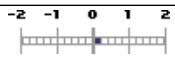
Respondents: 161 / Enrollment: 241

Summary: CS 1110-002 Introduction to Programming - Spring 2015 (17208)

Overall Course Rating

CS-1110-002 Mean 4.04
CS-1110-002 Std Dev 1.12
CS-1110-002 Response Count 803

Difference from Category Mean, Expressed in Category Standard Deviations

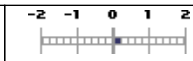


SEAS, 1000-level courses Mean 3.81
SEAS, 1000-level courses Std Dev 1.13
SEAS, 1000-level courses Response Count 6536

Overall Instructor Rating

INSTRUCTOR: Tychonievich, Luther
Mean 4.42
Std Dev 0.78
Response Count 1120

Difference from Category Mean, Expressed in Category Standard Deviations



SEAS, 1000-level courses Mean 4.13
SEAS, 1000-level courses Std Dev 0.99
SEAS, 1000-level courses Response Count 9240

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

1. 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 Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
161	4.57	0.63	102 (63.35%)	51 (31.68%)	6 (3.73%)	2 (1.24%)	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)
161	4.57	0.63	102 (63.35%)	51 (31.68%)	6 (3.73%)	2 (1.24%)	0 (0.00%)

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 Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
159	4.43	0.82	94 (59.12%)	48 (30.19%)	10 (6.29%)	6 (3.77%)	1 (0.63%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
159	4.43	0.82	94 (59.12%)	48 (30.19%)	10 (6.29%)	6 (3.77%)	1 (0.63%)

3. 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 Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
161	3.47	1.49	60 (37.27%)	28 (17.39%)	24 (14.91%)	25 (15.53%)	24 (14.91%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
161	3.47	1.49	60 (37.27%)	28 (17.39%)	24 (14.91%)	25 (15.53%)	24 (14.91%)

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

~
Question Type: Short Answer
~
contributed by Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Individual Answers
134	*See below for Individual Results*

Classes and program structure. It was interesting to see the how code can approach a large scale probleem.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The ask questions because I could learn anything
	Images
	The image stuff was cool to watch Prof. Tycho do, but really hard to learn to do myself.
	It's difficult to differentiate between different "topics" in the course, since it was cumulative and built upon itself. I guess I appreciated the nature in which things all came together towards the end of the course.
	I enjoyed the graphics lectures, as I had not previously learned any code using pixels.
	I liked learning how to design and program classes, because it makes my code much more organized and efficient.
	If-statements, because I love logic and this, to me, was one of the most important skills to learn and at the same time one of the least technical ones. It was easy to remember the layout of an if-statement. The important thing was to understand the logic behind them.
	Learning image manipulation the last few classes was my favorite.
	Reading files from the web
	Encryption was probably my favorite because I have always been interested in it.
	"Under the hood"
	Learning about the different classes
	just coding in general
	Creating classes
	recursive methods because they require thinking past the written code.
	If statements, one of the few things I understood.
	Learning all the methods and classes again was fun and a fresh reminder.
	Reading files and urls
	I greatly enjoyed almost every part of this course, but I specifically enjoyed working on the projects which had a larger end goal.
	Game design (Joust). It is what I am most interested in.
	Object Oriented Development
	Decryption
	cryptography
	I enjoyed learning about making new objects and creating classes. It came easily to me and opens up unlimited opportunities.
	Learning how to make classes because it was when everything started to make sense
	Image manipulation was fun because it was something I could actually use later on and have a function for.
	Image Manipulation
	Image Manipulation
	I could only follow the lectures on very basic topics
	ask tyco anything
	basic java, it helped me understanding what computer programming is and how it works.
	Image manipulation. Not as boring as other topics.
	I really enjoyed writing *simple* methods when we had a POTD that involved it. Some of the more complicated ones with Lou's List lost me a bit along the way, but I generally enjoyed learning how to make methods work together.
	Image manipulation because it has a lot of use in life

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Image manipulation, but honestly everything has been fairly interesting. Image manipulation is interesting particularly because it is so relevant (we are always dealing with pictures nowadays). The ones discussing while and for loops, thought that was cool I liked learning about class and methods because it was organized and broke complicated code down to make it easy to understand and write. I enjoyed the decoding lecture, where we had to use different numbers from different places around grounds to decode a message. I thought that this was both an enjoyable and effective learning experience. I liked classes/methods UML Diagrams - it was one of the only topics I fully understood. encryption&decryption Methods because it helped organize my code. I liked doing the email finder project (with regex) and working with data. I really enjoyed writing classes and methods, it was really rewarding figuring out what algorithm would work best to produce a certain result. Most interesting though were the image manipulation lectures because it's amazing to see yourself creating things that are actually common in our society and that you normally use another program to do. It was really cool! Class and method construction Learning to write code that interacted with the user was my favorite. image manipulation because it gave physical changes to changes in my code Making games was alright. The problem solving/algorithm. I found it fun and interesting. All of it!!! I enjoyed doing the game project because I got to apply what I learned I liked encryption the most, probably because I love codes and always have. None in particular. i liked making methods and learning how each small bit of code can come together to create a program Image manipulation. I loved to see how grayscale/color images were done. writing methods because it made a lot of sense to me I enjoyed writing classes because although the ones we did were simple, it helped me see the possibilities of CS. My favorite topic was Classes because it opened up a gateway for more complex java projects such as the game development and the email hunt. Chases The cryptology section because I am interested in encryption and decryption. Lou's list POTD's. I enjoyed learning about object oriented programming, because it is fun, practical, and easy to learn! image manipulation, it is the most useful and cool I had no favorite topic, but all of the topics were very useful! I thought making methods was the most useful thing because it gave me a better understanding of how computers work and I had a lot of flexibility in solving the problem. Game design, it was awesome to be able to create our own game. I liked learning about ciphers. making classes, it's the most applicable for the future

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Making classes because it was fun to make objects.
	Loops and if statements as you can do a lot of things with them!
	I really enjoyed creating our own classes and methods because it was cool to see how something I wrote myself enabled certain actions
	Method writing - challenging yet clearly useful.
	Creating Methods
	Image manipulation because it could be immediately applied
	Methods and classes
	array lists
	Classes - I understood them really well, and I think that's when CS started to "click" for me.
	Classes because it was more actual coding
	image manipulation because it dealt with so much that we had learned previously
	I don't have one favorite topic, I enjoyed the challenge and puzzle-like nature of the POTDS.
	I really liked learning to create classes. The uses seem infinite, and to me, classes represent one (probably simplistic) reason why computers can do so much.
	Recursion
	I really liked the unit on decoding hidden messages or sequences. I liked the problem solving aspect of it, and you knew when you got the right answer.
	Class writing was my favorite topic because it integrated everything we had learned in the class up to that point and made it easy to write bigger projects that did more complicated things much faster.
	I liked everything.
	I actually really enjoyed programming. I had a lot of difficulty with it since it was often very abstract, but putting it all together at the end was very satisfying.
	I enjoyed learning how to use loops in Java. This is a very fundamental concept, and we continued to use loops ever since for various different programs.
	My favorite topic was picture manipulation because it was really interesting and cool.
	Algorithms, because they are the fundamental concepts behind CS topics
	The Game project was as much fun as it was stressful. I (for the first real time)
	Creating loops and if statements.
	The Ask Tyconievich Anything Day
	Encryption
	The lecture that was my favorite was the one about turtles. It offered a simple way to make code and was a cool way to get students to learn how to write code
	I found the topic on image manipulation quite interesting because I was able to connect how this could be easily applied to various situations.
	I liked image manipulation
	I've always been interested in learning about computer science, so overall I enjoyed the entire course.
	paper airplanes...fun way to understand what he was talking about visually
	The encryption lecture.
	doing the Joust game
	Writing methods was my favorite topic because I thought it was interesting and I understood it well.
	Probably the different loops because their so useful.
	Creating classes because I could grasp how different classes were related and then I could make games and programs with the objects I created

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I liked learning about loops, I thought it was interesting when it was all new to me Learning how to make applicable code like email hunt I really enjoyed the topics regarding methods and classes, because it really helped improve my coding abilities and was very cool to implement on the final video game project. Under the Hood cleared up some big gaps in my knowledge hated learning about classes when i first started but when i got the hang of it i realized how useful they were! A++ Writing the POTDs I enjoyed the topic of game programming for the Game Project. I liked learning about reading CSV files The turtle project was my favorite. Even though I had no idea what I was doing, I was doing something and creating something fun! I was very proud of my drawing in the end and even showed it off to my friends. It looked like I was doing more than I actually was. Image manipulation because we actually got to look at things. The one about birds because I like birds. Encryption, because I found it very interesting. Decoding is very fun. Regressions, it was a difficult concept but made sense in the end. Working with games/graphics Loops, it was just logic and it was interesting Classes and objects. I had previously taken a computer science class and had struggled a lot with this topic in the past. I enjoyed the lecture about image manipulation because it made computer science applicable to real world scenarios. writing method Hard to say, I enjoyed all of the lectures. The lecture that does stick with me the most though is the one about the stack overflow where we got the demonstration of killing kitties. It was very entertaining and memorable! idk encryption was really interesting, I did not know very much about it Introduction to Java, It was the first topic, and only one that had appropriate time devoted to it. Making methods I liked loops because they made a lot of sense and they were fun to learn. The one where we learn about turtles cause it gave a good glimpse of what we were going to learn and it was fun. Writing and implementing our own methods; I felt like I understood this well and had the least difficulty with this topic uml diagrams, because of organization it provided. I enjoyed just learning to code. I liked learning about objects and classes. The one on picture manipulation. Learning methods is kind of fun
5. Which topic/lecture in this class do you think you will find the most useful in the future? ~ Question Type: Short Answer ~ *contributed by Tychonievich, Luther (lat7h)*	Results for CS-1110-002, Tychonievich, Luther Total: 132 Individual Answers: *See below for Individual Results*

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	All of it. Really just knowing how to start a project and lay out all the necessary elements.
	Class diagrams and design is particularly useful because they are not restricted to any one coding language, and can be implemented through lots of languages.
	If I were to continue on with CS, I think the lectures on classes and methods would be useful.
	I just think the general knowledge of how CS and the languages work will be the most valuable because i can apply that to other aspects of my major, which is math.
	I think debugging is most useful. You really have to develop your patience and problem solving skills.
	Probably all of it.
	making classes
	I believe just learning Java in general is very useful (cough cough Android cough cough) because the applications are endless!
	I think that I will find the methods topic useful.
	For/while loops
	Creating classes
	ArrayLists
	Reading files and urls
	The "Under the Hood" Section
	Learning loops because I will be able to use them in future subjects (commerce).
	N/A
	Image manipulation. I'm a cognitive science major so I won't necessarily work in computers but it has definitely piqued my interest and in something I can play with in my future. I hope to continue on and learn more so in that sense I presume learning methods/classes was most useful.
	Scanning projects
	All of them.
	I think for statements will be most helpful due to the logic it provides.
	Just the basic concept of dividing things into subproblems and dealing with them algorithmically. It's stuff I've done in Philosophy classes before and it was cool to see a different application of it.
	Objects, classes, and methods
	Algorithms, as an intended CS major I'd imagine that a lot of upper level CS courses will go much further in depth.
	Image Manipulation
	more focus on concepts and coding slowly
	Overall, learning how to come up with algorithms for writing programs was the most useful.
	Probably the ability to understand how code interacts with the computer.
	I think the ArrayList and images sections will be the most helpful in the future.
	None of them really. I don't plan on using CS in my job.
	Reading CSVs
	Writing classes
	All of it! CS allows us to think more critically/logically and will help me problem solve in the future
	Methods
	Methods
	Anything related to general problem solving strategies, how to approach a problem and come up with an algorithm or method to solve, and debugging is more useful for me than physically learning the syntax of typing out a code.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Everything
	Everything!
	Writing the POTDs, because it helped me build skills to write programs.
	all of them equally
	Definitely just the basics of coding (loops, conditionals etc.)
	how to use eclipse
	loops
	loops
	Class and method construction
	class creation
	Probably making methods
	Classes and Methods
	I think learning to create classes was the most useful topic.
	None.
	I think method writing will be the most useful because it is such a big concept.
	Learning how to design and program classes.
	the logic of coding & basics (ex. loops, if statements, methods)
	UML Diagrams
	for and while loops potentially while doing some programming as an intern this summer
	All of it, it was very helpful
	I found the general problem solving techniques used in CS the most useful. The ability to approach problems with logic and creativity to find an efficient solution is crucial in all fields, and I am grateful for this skill.
	Making classes.
	Writing classes and methods.
	game project because I am interested in programming videogames
	The lecture on methods seems like it will translate well into future CS classes
	I think making an algorithm performing math on inputs will be most useful to me.
	Basic programs that can do huge math problems
	beginning code
	The problem sets/labs where faulty code was given felt really helpful and applicable.
	understanding fundamental concepts
	Classes and methods.
	Honestly I thought that the lecture on where computers are and why was the most useful lecture. You learn to appreciate the importance of programming, and the benefits of having computers in our lives.
	I think I will find the basic logic behind many of the operations useful as I move on in computer science.
	The graphics segment.
	Java
	Recursion
	Using arrays and arraylists

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	General programming ability
	the game project and email hunt
	...everything...?
	Ask me anything
	basic cs knowledge
	none
	The lectures on building classes will be very useful going forward.
	Opening/reading files and websites
	Encryption
	I think the most useful topic and lecture were the ones about Classes and methods.
	All of it.
	The fundamental ways of building an algorithm.
	building classes and methods
	the same one as above
	The way of thinking in general. It has given me a new appreciation for step-by-step problem solving, which I am sure will never stop being useful.
	Probably class writing, because no matter what sort of project I want to do, I would be able to write it succinctly.
	being able to write methods
	Reading CSV files and Arrays/ArrayLists
	All of them -- I'm going to major in computer engineering.
	I think the fundamentals will be most useful.
	looping
	loops if
	if statements
	classes and learning to carefully debug and patience lol
	I think the concept of writing these small classes that do simple tasks for you is pretty useful. I'm definitely decided on not being a CS major after this class, mostly because I find it a pretty stressful subject, but I like that I have a basic understanding of how programming works now. I think this'll help when I have to learn MATLAB in the future.
	Probably the class time spent learning about arraylists and sorting them
	The one with the help coding.
	literally all of it, everything we learned seems like it will be useful, great framework for beginning programming
	All of it. I had no prior experience, so all of my skills acquired will be helpful.
	Method writing
	All of it
	All of it
	I think everything in this course would be extremely useful in the future, I just struggled a lot with all of the topics.
	interpreting code in languages we have not seen before.
	Understanding java as a whole, possibly also the logic of image manipulation
	none of it.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Reading and understanding code in Java. The logic aspects of CS and the way it makes me think I think everything I learned in this class will be useful the future because I am planning on majoring in CS. I'm completely new to computer science so it's hard for me to say. I feel like everything was vital and extremely useful. Basically everything. Classes/Methods The lecture(s) that were most useful were using and learning how to make loops Introduction to Java, I now have a basic understanding of coding, but know that I will need to take another intro level CS course to understand it later in life. Problem solving/algorithms. the ability to avoid ambiguity. None in particular; I don't intend to take another CS course All of it seems useful. All of the topics seem useful! The use of classes and algorithms. Methods. Making methods Sorting algorithms because they are uni-language I believe the lecture that will be the most useful was the lecture about writing classes because it incorporated everything we had learned in the course. just the basics of programming I think the methods were really useful as they encompass many aspects of CS. just the basic coding learned at the start of the semester Classes because I'm assuming that is how most programs work Probably writing methods I believe all topics will be useful since this is the basis for computer science.
6. What lecture/topic(s) in this class "did not work" or were not seen as useful in the long run? ~ Question Type: Short Answer ~ *contributed by Tychonievich, Luther (lat7h)*	Results for CS-1110-002, Tychonievich, Luther Total: 123 Individual Answers: *See below for Individual Results*
	N/a All were useful. Almost the entire thing. The pace of this course is far too fast for someone who has zero experience with programming, very few concepts were adequately explained in lecture, and the homework never reflected what was actually taught in lecture. i can't think of anything that didn't really work. I felt as if the majority of the lectures were a complete waste of time and were not relevant to our assignments or did not adequately teach the class how to do the assignments. I thought the initial few classes on Turtle were personally for me not particularly useful. Although I do understand the point of it as an introduction to Java and Eclipse, I think there would have been better ways to do it.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	There was too much homework.
	The loops at the end disappointed me because I really wanted to learn cyber security.
	I think the introduction of the game project could've been very different... I found the project to be the absolute most stressful one I've done in my whole first year at UVA. I think that project is more one for students who have an interest in majoring or minoring in CS instead of those simply fulfilling their first-year E-school requirement. I don't feel like I have the brain for computer science, so I spent entirely too much time on the project trying to figure it out.
	Nim was dumn
	They were all very relevant and useful!
	The image manipulation classes were a little confusing.
	I liked all of the topics
	nothing.
	N/A
	N/A
	N/A
	N/A
	Searches and sorting
	-
	-
	I'm still not quite there with image manipulation.
	I wasn't particularly excited about image manipulation because I'm not a visual person, but it was alright.
	I think arrays are not important. I understand why we need to know about them, but we're never going to use them in the future.
	Image Manipulation
	Image Manipulation
	anything taught by a TA
	I'm having a hard time caring about how to visually manipulate images.
	They all worked well.
	The second scavenger hunt was too large an undertaking for a single class
	the ones taught by TA's
	The arrays were never used past an introduction.
	Drawing images
	The game project had too much material that we had not learned about, which made it frustrating occasionally. Overall it was a good experience, but I wish we had learned more about graphics, etc. beforehand.
	None of them really.
	Turtle drawing
	None
	None
	None
	None
	I did not really like the special topic of image manipulation at the end of the year. it was interesting, but not really applicable at all. i think this is due to the fact that we had to keep it very basic because of the advanced concepts it requires.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	None. Everything seemed helpful and relatively important
	image manipulation because we did not learn enough about this to have it be useful in the long run
	I thought all lectures were interesting and useful
	The part about light, and the beginning lectures using turtle and world
	The class about searching and sorting felt irrelevant. Also any class taught by a TA was not interesting.
	image manipulation seems very specialized and not that useful
	None.
	I thought that each of the topics were significant in the long run. Nevertheless, there were definitely some topics that were more difficult than others.
	The first couple of lectures were useless and went completely over my head.
	The big array list questions like library and lous list were really tough for me to understand
	CSV file reading
	Sorting explanations seemed to take up an unnecessary amount of time
	Image manipulation
	Image manipulation
	nothing
	The stuff about the stack, activation record, and the specific parts of the memory system are not useful for any non-CS major who isn't concerned about the intricate workings of computers.
	I had the most issues with file opening/reading, but that doesn't mean it won't be useful in the long run. I can't think of anything that won't be.
	none i can recall
	None
	Lectures that required us to do code in class.
	Arrays.
	N/A.
	I think the flappy bird lab and the final game project were slightly inapplicable because they have little to do with algorithms and problem-solving
	I did not like Image manipulation, I did not understand it as well so it will not be as useful
	Maybe reading files and websites? I know they will be useful for some people but I don't see myself ever using them again in the practical or recreational sense.
	none really
	Everything I think is useful, but if I had to pick a least useful I would pick image manipulation
	Recursion
	The painting with turtles was kind of useless in my honest opinion. It was still fun though.
	most of the rest seemed to not cover any pertinent new information
	Recursion
	blurred image
	Can't think of one.
	I did not think the POTD Nim was useful for understanding nor would it seem to be very useful logic for the future.
	none
	none

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	none
	none
	none
	Most topics were useful.
	The lectures about creating pictures - very confusing
	decryption
	Nim
	Nim
	For me graphics and images are not very useful
	I didn't enjoy the image manipulation part of the course. It wasn't really relevant to anything we were learning beforehand and just came out of the blue
	recursion
	recursion
	image manipulation
	coding slowly
	a lot of the chases and encryption
	When the TA taught the lecture it was more difficult to learn.
	Image editing
	Imaging, scavenger hunts during the class period.
	The paper airplanes
	a lot of lecture was pointless and i found myself teaching myself most material
	Encryption is probably the topic that is the least useful in the long run.
	special topic
	It seems like we just kind of wizzed through the search algorithms unit. It might've been more helpful if we did 1-2 POTD's on search algorithms to get a better understanding on how they work
	Encryption.
	Recursion "did not work" for me in particular because I never grasped the concept of it, but I could see it being useful in the long run.
	nothing in particular, but i felt like we spent way too long on string manipulation. we could have covered the topics much faster.
	reading files from the internet or from the java package, I could never figure out how to completely figure out how to do it
	Picture manipulation
	I think that everything built off of each other pretty well. Again, I'm not the best at it, so some things didn't make too much sense.
	I think the lecture and topic that needed improvement were the beginning lectures on classes and methods, I understood them later on but at first when they were introduced it was a bit confusing.
	One topic that gave me a lot of trouble was the one about methods. I didn't fully understand it until we did that show and tell lecture where we had to actually make methods for what we had. I can't clearly remember the initial lecture about methods so I can't give specifics but I feel like that one was the only one that wasn't as affective.
	the whole course in general----the logic of coding & basics (ex. loops, if statements, methods)
	None of it.
	None really most every concept is useful in some respect. I can't say that about too many classes.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

I think all topics are useful in the long run, but I am having trouble learning how to use the image manipulation techniques in a practical manner.

These last few seemed a little superfluous.

Image manipulation mainly because I wanted to learn about encryption machine learning. It seemed like image manipulation was just different applications of material we had already learned, but maybe that's that point.

Not sure if image manipulation will be of much help in the future.

Cyphers and Image Manipulation

turtle

I feel that recursive methods was the least useful topic as not many people will use such a complex method.

None of the topics stood out to me as not useful.

sorting and searching

Nothing really

The Nim POTD

Not sure

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

~

Question Type: Likert

~

contributed by Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
161	3.58	1.19	39 (24.22%)	58 (36.02%)	35 (21.74%)	16 (9.94%)	13 (8.07%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
161	3.58	1.19	39 (24.22%)	58 (36.02%)	35 (21.74%)	16 (9.94%)	13 (8.07%)

8. How would you rate the availability of TAs?

~

Question Type: Likert

~

contributed by Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
161	3.17	0.75	60 (37.27%)	69 (42.86%)	31 (19.25%)	1 (0.62%)	0 (0.00%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
161	3.17	0.75	60 (37.27%)	69 (42.86%)	31 (19.25%)	1 (0.62%)	0 (0.00%)

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

~

Question Type: Multiple Choice

~

contributed by Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Every week (NA)	Every other week (NA)	Once per assignment (NA)	Rarely (NA)	Never (NA)
161	16 (9.94%)	27 (16.77%)	13 (8.07%)	54 (33.54%)	51 (31.68%)

Results for SEAS, 1000-level courses

Total	Every week (NA)	Every other week (NA)	Once per assignment (NA)	Rarely (NA)	Never (NA)
161	16 (9.94%)	27 (16.77%)	13 (8.07%)	54 (33.54%)	51 (31.68%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

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

~

Question Type: Likert

~

contributed by Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
160	3.09	0.82	53 (33.12%)	75 (46.88%)	26 (16.25%)	5 (3.12%)	1 (0.62%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Excellent (4)	Good (3)	Average (2)	Weak (1)	Very Poor (0)
160	3.09	0.82	53 (33.12%)	75 (46.88%)	26 (16.25%)	5 (3.12%)	1 (0.62%)

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

~

Question Type: Short Answer

~

contributed by Tychonievich, Luther (lat7h)

Results for CS-1110-002, Tychonievich, Luther

Total	Individual Answers
103	*See below for Individual Results*

Graham Turner, Justin Dao, Scott Mallory, and Casey (forget-her-name) were amazing TA's. Very useful and helpful for assignments, but also for explaining concepts. I would highly recommend asking them to return, unless they are graduating.

Some were very helpful and some were not

They are very knowledgeable and understand where the problem in my understanding of a certain topic is which makes resolution of the issue quick and to the point.

no

no

no

Scott and Jasmine are awesome!

I asked TAs questions mostly during our lab sessions or during reviews before tests. While they all knew what they were talking about, some didn't know all the details of how to put something into code, and ended up confusing me more than helping me. There were also great TAs, but they were generally harder to get to when there were questions because other students also preferred asking them.

Eric Siegal was very helpful

very enthusiastic

They were more helpful

Stephen and AJ were our lab TAs. They were very helpful

Some were better than others

From my understanding the TAs struggled to answer and help fix basic problems in code for many of my peers.

The TAs were always helpful, but there were often a lot of people and so I could only be helped very briefly.

Some of them are really terrible! They're straight up like "uhhhhh try fiddling with this a little bit more and see if it works" like i knew that, that's not why i'm asking you to help me. But some of them are lovely! Help to debug my code and stick with it until they help me get a step further or fix my problem.

Nope

Nope

none

none

none

They were amazing!

The TAs were hilarious and they knew the subject matter, and it made me really enjoy lab.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	The TAs did a great job explaining concepts in office hours and during lab each week. It was nice having someone to talk problems out 1 on 1.
	A bunch of TAs from whom I've gotten help were consistently extremely helpful. Jasmine, Madelyn, Scott, Steph, and Jim were always a HUGE help when I needed it. I'm sure all of them are fantastic, but those were the ones I ended up getting from the queue most often when I showed up for office hours.
	nah they are good
	There remains an expectation that students know something about CS despite the intro level and description of the course.
	They were nice and tried to help but often assumed I knew too much
	Very helpful and availability
	TAs were good in lab but I was never able to go to the office hours because I could not commit to waiting as long for help as other people had said that they waited.
	The TAs in lab were helpful.
	I like my TAs for my section, Casey and Justin
	There were some TAs that were more helpful than others. Some didn't even know what the POTDs were which made it difficult to ask questions, but overall they were really helpful and seemed to know their stuff! I preferred going to the office hours rather than posting on Piazza because the response times weren't too quick and the TA's response on Piazza were just short and often didn't answer the question I was trying to ask.
	CANNOT SAY THANK YOU ENOUGH TO THE AWESOME TAS :)
	Mostly very helpful. I loved being able to just go to the "general" office hours and get help!!!
	They shouldn't teach lectures.
	N/A
	N/A
	N/A
	N/A
	Justin and Casey were awesome and super helpful.
	Less helpful as the year went on
	The TAs were extremely helpful, both in labs and office hours.
	Stephen and AJ were great in lab, Casey was very helpful at office hours.
	The TAs were often stretched and had little time to help but were extremely helpful when they had the time.
	-
	Some were particularly condescending
	When I needed help, they found a way. Very tenacious.
	Some of the TAs rushed in helping me during office hours to get through the OH queue
	They are worn too thing, do not have enough time to help. You wait hours to see them and then they give you no assistance.
	Scott and Jasmine were good TAs!!
	they were great, that is all
	TAs are kind and cute
	There were some TAs who were very helpful. Some others were very unhelpful. They wold tell me to do one thing, then walk away. I would do what I thought they had told me to do, but that would not work, and they would not come to check to see how I had progressed in the assignment.
	Very kind and willing to help, just not always available.
	David is the man.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Some were more helpful than others who you could tell were just trying to get through as many people as they could even though the list wasn't long. I didn't learn any names so I can't give any in particular. However, for the most part they were extremely helpful and wouldn't judge us for asking "stupid" questions.
	Jim (the one who wears the Microsoft shirts) is the #1 TA
	The TAs in lab were very useful and helpful. They were willing to work with us and fully answer our questions, and I am appreciative of their assistance.
	My TA's from section one of lab were very helpful and related well to the class.
	Overall, the TAs were always available to help and answer questions. I found that Piazza was a very valuable resource as a means of getting questions answered and doubts clarified. The TAs were all very knowledgeable and were adept at explaining intro CS concepts.
	Some TAs really didn't seem like they knew anything on the other hand, some TAs were very very good (like the ones in my lab.. lab section 104)
	They were a great deal of help and made themselves available.
	Lots of people; very little amount of TAs. TAs were careful not to give us the direct answer or method of coding for our questions, but sometimes did not point us at all a clear directions.
	They are all very helpful, regardless of whether or not you were in their section.
	While they were great, office hours can get crowded very fast. Although they were very helpful on Piazza and were quick to respond to that.
	The TAs' advice was so vague that it never helped me understand things better and I basically had to learn everything on my own.
	They were sometimes unhelpful by simply asking me what I asked them..
	TA office hours were always really overcrowded.
	The TAs were helpful if I had to use them, which was rarely.
	some would point you in the right direction, while some would just fix it for you. it was kind of luck of the draw.
	My TAs were wonderful.
	None
	None
	My only interaction with them was in lab (105) but they were always available and helpful.
	Scott Mallory was awesome
	The TAs were an indispensable resource who really made me think hard about the problems I was facing to see if I can reach an solution myself without it being given to me.
	Very helpful! Very patient
	Stephen and AJ were really nice and helped anytime I needed it
	they seemed very overworked to me
	None.
	None.
	Some of the TAs were amazingly awesome and genuinely wanted to explain the material to me and actually spend the time helping me with my code. Other TAs were complete assholes and very rude/condescending/absolutely no help and showed no interest in helping. Other times the TAs seemed to struggle with how to solve some of the POTDs/Projects.
	Justin was an AWESOME TA...extremely helpful and great to work with...very clearly passionate about CS as well
	By the end there just weren't enough to support the need.
	They were helpful on Piazza, especially Matt Pearson Beck
	They were very knowledgeable and knew how to help without giving us the answer.
	No

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	n/a n/a n/a Jim was great! They were really helpful and knew what they were talking about. not really I thought TAs Stef, Jas, and Justin were especially helpful, although I don't know much about the other TAs. they were approachable and knowledgeable I love that the TA office hours were the same time each day. That really helped. The TAs are different in the amount of help they give, though. Some are helpful, others less so. Some TAs seemed "to busy to help" at times Some were better than others I asked questions via Piazza and I could always count on a TA helping. Allison Moyer was excellent as a TA. Very helpful. Pretty helpful They were smart.
12. What would you suggest we change about this course in the future? ~ Question Type: Short Answer ~ *contributed by Tychonievich, Luther (lat7h)*	Results for CS-1110-002, Tychonievich, Luther Total: 103 Individual Answers: *See below for Individual Results*
	Spend more time coding in the first couple weeks so we can go slower throughout the year. no Not much. I'm not a huge fan of the last unit that we're doing (photos). I think that the four hour delay on the feedback from grading system is too long to be useful in many scenarios. I would suggest cutting it down to 3 or even 2 hours. More labs/potds that teach us about image manipulation and joptionpanes More about why we use different objects/methods/etc, rather than just when to use it. i.e. usefulness of each thing Maybe have practice tests posted. listen to your heart I liked this course the way it was less hw assignments, make the class worth 4 credits Advertise it as 'having experience is very helpful'. Not much... learned a lot. Nothing Add more TAs. It's great as it is. more TAs at busy office hours Eliminate the lab portion because it wasn't very helpful.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Perhaps break up the course into a smaller class size, at times 300 students seems like a lot for one course. Aditionally, maybe transition to teaching intro programming in Python, as I have heard that it is easier to learn than Java.
	make the program of the days easier or give feedback quicker
	N/A
	-
	Don't let TAs teach lectures, don't do in-class assignments (encryption chase, show=-and-tell, etc.), labs seem pretty irrelevant sometimes.
	Give more guidance as to what you are testing for with Programs of the day. There were several times where I received errors for test cases not in the examples and I could not find where the resulting error would come from. A hint or clue as to where they were would be more conducive to learning than just telling me that something random went wrong.
	Don't do the partner projects
	the amount of time to complete POTD when it gets to the end of the semester, as they get more difficult, and the time spend on them increases dramatically. Therefore it was some what unrealistic to get them all done every two days. i ended up turning things in late a lot.
	offer the option of whether or not to participate in pair programming
	The difficulty of some of the POTD's.
	We should learn markup languages! And learn how the command line programs we made could be useful, or just something cool that computer science could help us with in our everday lives.
	It would be helpful if the professors could go slower in terms of programming - the fast typing and explanations made it difficult for beginners to process and follow along during the lectures.
	The tests seemed very hard compared to course material.
	Can't think of anything
	This course is too difficult for an introductory course
	I can't think of anything that would be efficient to implement
	the coding activities in class doesn't really help on the materials on the exam (the labs are much more complicated)
	choose better TAs?
	I found that many of the assignments included a lot of code that I feel wasn't covered in class. In addition I think it is ridiculous that a POTD would lose points for missing a single comma or misspelling in a printed statement. In addition,
	I didn't like the pair programming, because it caused me to only understand part of the code. I learn by doing much more than by watching, and would have appreciated being pushed more to figure solutions out on my own rather than watching someone else do it. For the parts of the pair projects that I came up with and explained to my partner/put into code I found these assignments very useful. So I would suggest not changing the assignments, but giving the option to work alone.
	Less time passing before seeing automated grading system results.
	None
	I would advise that the course be more organized towards the end of the semester, since I got a little lost.
	At a certain point in the semester, an overwhelming amount of students seemed to always be at the office hours. Obviously TAs have other things going on, but if there was an effective way to help each student it would be very beneficial.
	Make sure that all the CS courses cover the same material as Sheriff's lecture, specifically relating to pass by value concepts.
	It's a little bit harder than I thought even though I had some past experience in programming.
	Change TA office hours so you don't have to sit there for over half an hour to see a TA and spend hours at office hours just to get sufficient help. There should be group office hours (for example, there could be a review session of sorts for the start of every POTD in which a TA could give a general outline of the problem/solution).
	None.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	SLOW DOWN this course is way too fast, the homework is way too complicated and the lectures are inadequate preparation for assignments.
	n/a
	n/a
	make it four credits
	Get rid of Nim
	Nothing it worked pretty well
	change to python
	Go slower through material
	It moves too fast: I enjoyed it at first, but after a few weeks I had no understanding of anything the professor said.
	Give individual larger projects. For example POTDS that lasted two weeks and required more work.
	Slow down the pace, a lot. I fell behind towards the end of the class.
	Spend more time on the fundamentals before the first midterm and not jump right into it at the beginning of the course.
	Maybe just more material to review/practice codes for tests
	Don't change anything please!
	python
	Giving students the tools to help them figure out what is wrong with there programs.
	After having finished a program, how does someone package it so that it can be shared and run on other computers without eclipse?
	speed still a bit fast
	I did not like the Show and Tell class. I am OK talking with random people on occasion, but an entire class devoted to working in groups when I expected a lecture was not good.
	A class before this one to introduce more fundamental concepts about CS
	none
	Provide solutions to the in-class hunts and things after the due date has passed. I think I could have learned a lot about particular subjects from that.
	Can't think of anything.
	maybe more debugging practice would be helpful, it is not fun or easy, but I think it is helpful
	Make more sections for students without prior programming experience
	Learn cyber security.
	Perhaps some sort of class participation aspect, maybe reflected in grading
	Use Panopto to record what's going on on-screen, as well as the audio. This makes it much easier to catch up on material when we miss class.
	Different special topic at the end of the semester
	Way too much homework...I struggled with this course a lot, so the homework would take me hours at a time. It was way too time consuming.
	Fewer POTDs or ones that are bigger, but more spaced out.
	This has literally been my favorite course taken at UVA and it's not even my major so don't change a single thing. Actually, more poems would have been nice.
	N/A. Good luck with Python.
	This class was honestly pretty darn good.
	POTD directions can be more straightforward and less lengthy. Plus, when instructors answer questions in Piazza, please do not include a link to another's question and that's it. Answer it again.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Teach over a broader range so we can learn more applicable things. The first couple of months didn't go deep enough to be helpful after the first week. Maybe deal with more complex algorithms, bring in more outside methods and classes, or focus on teaching the class better. I don't think there's anything to change. I think the course is appropriate. However, sometimes what is gone over in lecture is not described well in the collab notes from lecture. That sometimes makes it really confusing when going back and doing homework. Not all labs were conducive to pair programming and were ultimately frustrating on that accord. I think the course got very difficult very quickly and if the material didn't make sense to someone it was too easy to get behind. More web development stuff, talking about Internet. nope Nothing, it was great. Make the POTD grading appear faster: perhaps 2 hours instead of 4. Either have more TAs or a better system for helping students through the projects...as is it seems like too much of a trial by fire this course was definitely aimed towards students that have already taken CS. It was very difficult to keep up and the instructor would make leaps difficult to follow from the perspective of someone who has never had any experience in CS I'm only taking this for the Math major's requirements -- maybe a section specifically for non-CS people? It's probably not practical, but it's the only improvement I can think of. Make it more accommodating for students who have never had any exposure to computer science before. Such as have extra practice outside of lecture and lab focusing on the basics to help those students catch up with everyone else. Should include less homework My only suggestion is to take the course a little slower. I feel like the last week or so has been "filler" time, especially with the image chase scheduled for tomorrow. I think the extra time could be spread out over the part of the course dealing with methods and classes. For me, at least, with absolutely no programming experience coming into the course, it was pretty difficult (understatement) to transition from writing everything in the main method in a simple fashion to writing full-blown methods and classes in a matter of a week or two. It didn't make sense in my head for a good month after we covered it. (I probably should've taken CS 1112, but in the end I'm glad I took the more challenging one for beginners.) The homework for classes and methods weren't helpful as much as the other home works. i don't need 4 hours as incentive to test my own code. i like the time delay idea, it's good. 4 hours is too long. I think the tests could have been less in depth with certain topics and more focused on assuring that we can apply lessons taught in class to write our own program. I also don't think hand writing code is entirely necessary. Get rid of the 4 hour delay in submitting POTDs, we only have 2 days to do it so it's hard to see if you did something wrong before it's too late, especially near the end when it was harder to "break your code" Nothing - very effective This should be a 4 credit class. I think it was very well structured! Partner work is always a little hard because people don't necessarily pull their weight or take your ideas into account, and although I know we fill out evaluations for them later it still makes the process of completing the project difficult and risks the product. Make it so I can partner with someone outside my lab. The restriction of working with someone within my lab was quite annoying.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

13. 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)
159	4.36	0.71	74 (46.54%)	72 (45.28%)	10 (6.29%)	2 (1.26%)	1 (0.63%)	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)
1305	4.06	0.93	446 (34.18%)	611 (46.82%)	155 (11.88%)	54 (4.14%)	36 (2.76%)	3 (0.23%)

14. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
161	4.29	0.85	75 (46.58%)	65 (40.37%)	14 (8.70%)	3 (1.86%)	3 (1.86%)	1 (0.62%)

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)
1325	3.93	1.14	480 (36.23%)	434 (32.75%)	171 (12.91%)	100 (7.55%)	64 (4.83%)	76 (5.74%)

15. 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)
161	3.99	1.21	68 (42.24%)	60 (37.27%)	7 (4.35%)	15 (9.32%)	11 (6.83%)	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)
1310	3.94	1.07	433 (33.05%)	582 (44.43%)	127 (9.69%)	102 (7.79%)	59 (4.50%)	7 (0.53%)

16. 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)
161	4.48	0.69	92 (57.14%)	59 (36.65%)	6 (3.73%)	4 (2.48%)	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)
1306	3.95	1.14	478 (36.60%)	450 (34.46%)	160 (12.25%)	88 (6.74%)	70 (5.36%)	60 (4.59%)

17. 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)
161	3.25	1.30	26 (16.15%)	18 (11.18%)	31 (19.25%)	21 (13.04%)	11 (6.83%)	54 (33.54%)

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)
1308	3.19	1.21	132 (10.09%)	185 (14.14%)	257 (19.65%)	124 (9.48%)	87 (6.65%)	523 (39.98%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

18. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
158	4.19	0.89	66 (41.77%)	67 (42.41%)	17 (10.76%)	5 (3.16%)	3 (1.90%)	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)
1320	3.99	0.99	413 (31.29%)	580 (43.94%)	161 (12.20%)	61 (4.62%)	46 (3.48%)	59 (4.47%)

19. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
159	4.81	0.41	129 (81.13%)	29 (18.24%)	1 (0.63%)	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)
1319	4.47	0.75	723 (54.81%)	425 (32.22%)	71 (5.38%)	13 (0.99%)	14 (1.06%)	73 (5.53%)

20. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
161	4.53	0.69	101 (62.73%)	48 (29.81%)	9 (5.59%)	3 (1.86%)	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)
1320	4.39	0.78	653 (49.47%)	474 (35.91%)	75 (5.68%)	25 (1.89%)	14 (1.06%)	79 (5.98%)

21. 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)
161	3.64	1.33	26 (16.15%)	21 (13.04%)	14 (8.70%)	8 (4.97%)	8 (4.97%)	84 (52.17%)

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)
1307	3.60	1.19	199 (15.23%)	238 (18.21%)	179 (13.70%)	79 (6.04%)	53 (4.06%)	559 (42.77%)

22. The grading policy was fair.

~

Question Type: Likert

~

contributed by Dean of the School of Engineering and Applied Science

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
160	4.12	0.94	63 (39.38%)	68 (42.50%)	17 (10.62%)	9 (5.62%)	3 (1.88%)	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)
1317	3.77	1.15	375 (28.47%)	483 (36.67%)	198 (15.03%)	128 (9.72%)	72 (5.47%)	61 (4.63%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

23. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
161	4.46	0.69	87 (54.04%)	62 (38.51%)	9 (5.59%)	1 (0.62%)	1 (0.62%)	1 (0.62%)

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)
1321	4.17	0.90	494 (37.40%)	535 (40.50%)	126 (9.54%)	46 (3.48%)	24 (1.82%)	96 (7.27%)

24. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)	Not Applicable (NA)
160	4.58	0.63	102 (63.75%)	50 (31.25%)	6 (3.75%)	2 (1.25%)	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)
1318	4.19	0.97	570 (43.25%)	456 (34.60%)	124 (9.41%)	57 (4.32%)	32 (2.43%)	79 (5.99%)

25. 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)
161	1 (0.62%)	34 (21.12%)	80 (49.69%)	31 (19.25%)	15 (9.32%)

Results for SEAS, 1000-level courses

Total	Less than 1 (NA)	1 - 3 (NA)	4 - 6 (NA)	7 - 9 (NA)	10 or more (NA)
1311	168 (12.81%)	507 (38.67%)	422 (32.19%)	149 (11.37%)	65 (4.96%)

26. 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)
161	4.46	0.70	90 (55.90%)	58 (36.02%)	10 (6.21%)	3 (1.86%)	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)
1305	3.95	1.11	500 (38.31%)	472 (36.17%)	163 (12.49%)	114 (8.74%)	56 (4.29%)

27. 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)
160	4.41	0.86	92 (57.50%)	50 (31.25%)	12 (7.50%)	3 (1.88%)	3 (1.88%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
1308	3.81	1.29	509 (38.91%)	394 (30.12%)	172 (13.15%)	113 (8.64%)	120 (9.17%)

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~

28. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
157	4.51	0.57	86 (54.78%)	65 (41.40%)	6 (3.82%)	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)
1317	4.11	0.93	491 (37.28%)	605 (45.94%)	138 (10.48%)	46 (3.49%)	37 (2.81%)

29. 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, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
161	3.93	0.93	49 (30.43%)	64 (39.75%)	37 (22.98%)	9 (5.59%)	2 (1.24%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
1324	3.82	1.03	382 (28.85%)	495 (37.39%)	314 (23.72%)	91 (6.87%)	42 (3.17%)

30. Overall, the instructor was an effective teacher.

~

Question Type: Likert

~

contributed by Office of the Provost

Results for CS-1110-002, Tychonievich, Luther

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
160	4.25	0.87	74 (46.25%)	61 (38.12%)	18 (11.25%)	5 (3.12%)	2 (1.25%)

Results for SEAS, 1000-level courses

Total	Mean	Std Dev	Strongly Agree (5)	Agree (4)	Neutral (3)	Disagree (2)	Strongly Disagree (1)
1327	3.96	1.08	483 (36.40%)	507 (38.21%)	203 (15.30%)	68 (5.12%)	66 (4.97%)

31. 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
67	*See below for Individual Results*

It made me consider pursuing CS further.

Tychonievich is the man, he was an awesome professor and his teaching style is great and helpful

fun class, but it was too slow.

Tychonievich is so knowledgeable, passionate and enthusiastic about CS, it was really exciting and engaging to learn from his lectures and was definitely one of my favorite courses here at UVA.

Good course overall, but I didn't really enjoy lab.

Definitely better than I expected it to be, I thought it really made learning CS possible/approachable/less intimidating

Good course, hard material. Sometimes the POTDs were a huge leap ahead of what we learned in class which made them very difficult. It was like being taught to make a PBJ and then being thrown into a 5 star restaurant kitchen.

Not very friendly to new CS students. I was at a severe disadvantage the whole course. A lot of students had a lot of experience in CS already and the class really catered to them more.

Worthwhile course, but a very fast pace

This course was very helpful and was a good introductory course for people with no prior computer science experience.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I really liked this class, but I think that whoever teaches the course should make the test because they would focus on what is actually asked on the test.
	Good course, will help me a lot in future courses, sometimes the assignments were intimidating
	Luther is bae
	I'm a math major but this class was my favorite class I have taken at UVA. It was surprisingly interesting and extremely informational. The instructor was also my favorite instructor that I have had so far. He was very helpful when I would go to his office hours and you can tell he's a professor that genuinely wants his students to learn all they can and actually enjoys teaching. THE BEST!
	I loved the course and found it very helpful. I am also taking CS 2102 (Discrete Math) this semester and have noticed that they work well together and each one helps me understand material of the other one. I would suggest making them co-requisites in general rather than having CS 1110 as a pre -requisite for CS 2102.
	the professor is very energetic and i really like it.
	While I admit that computer science is not my strong suit, I think that in my experiences this semester I have come to the conclusion that the subject could have been taught in a more effective manner.
	The class was taught very well and made me enjoy CS
	Overall (though I am writing this before the final), I think this is a worthwhile course that's a good introduction to programming in the engineering school. While I do think it should require less as an *intro* course, it was interesting material to learn. I guess some people are CS-brained, and I am definitely sure I'm not one of those people now.
	N/A
	N/A
	Good intro course for CS.
	Good course - no changes needed.
	I think Professor Tychonievich is very knowledgeable, but sometimes he went way too fast in lectures to the point where many students were lost. Also, the course moved so fast that it was nearly impossible to catch up if you fell behind for any number of reasons.
	-
	It was my favorite class!
	Good course. Jim is a great TA.
	I was thoroughly disappointed with Professor Tychonievich. He just did not teach in a manner that was conducive to learning the material well. LOTS of other students had similar problems with him. I kept hearing about Sherriff's class having an easier time, or getting more help/hints with homework assignments. After the first test, which I did fairly poorly on, I started attending Sherriff's lectures instead and noticed a drastic difference in how well the material was being taught. My grade went up over 20 points for the next test. Additionally, Tychonievich missed about 6 of his lectures, having a TA teach instead. That is unacceptable. Students are not paying such high tuition to have a student teach them when we have such a successful CS department. This also made it feel as though Tychonievich did not care for his students, since he so often missed class. Overall, I'm glad I took CS 1110, but only because Sherriff quickly changed my mind. This was not at all thanks to Tychonievich.
	Good course, definitely could teach classes and methods better but that's about it.
	PLEASE. LET. DR. TYCHONIEVICH. STAY. FOREVER. HE. IS. TOO. GOOD. TO. LEAVE.
	Too hard for an intro level class!
	Very interesting
	He could have been better. The lectures weren't that great. Many people stopped going because they found them useless. I went everyday but didn't get as much out of them as I wanted to.
	The class is far too fast paced and expects students to have some background in CS, despite explicitly saying students should not have a background in CS when taking this class. The grading system is too harsh when considering the complicated nature of the assignments compared to what is in lecture. Anyone who has not already taken CS should not take this course. This class is without out a doubt the worst class I've taken at UVA.
	Good class
	Professor Sheriff and Tych. are both very humorous but IMO Professor Sheriff is more interesting thus we learn the information better.
	sweet class

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	Professor Tychonievich was, by far, the best professor I have had at UVA thus far. Not only was he very knowledgeable on the subject matter, but he was obviously very passionate about CS, which made his lectures even better than they already were. I was tentative about this course at first (since I had no prior experience whatsoever), but with Professor Tychonievich's excellent teaching abilities, I picked up the skills very quickly, allowing me to excel in the class.
	It's a very good course! I just wish you had shown us some ways to make us fall in love with coding more. Like, it would be great if I could make something with coding that would affect my everyday life and make it better.
	Just my opinion, but I feel like CS involving computers would make a far more effective test than paper tests. I just don't feel that normal programmers would ever code in paper under pressure unless they are writing pseudo-code which we are not. I'm also pretty sure that having online tests are not that hard for CS.
	Good course, convinced to major in computer engineering!
	Good class, I liked the problem solving aspect of it!
	An excellent and fun course to take overall
	The POTDs were demanding and took up too much time weekly, though they were definitely extremely helpful in my learning. It would be nice if the grading system timing was shortened though, 4 hours is too long to wait, especially when working on a tight schedule.
	hooray
	None
	None
	This course is best suited for students who have taken CS before. It is not very suited for students who have had no prior exposure to CS.
	Great Class!
	Very well structured
	It was great.
	Luther is a wonderful, spritely individual
	My primary criticism of this course was that it was labelled an introductory course. The only students I know of who found the material manageable and fair had some sort of previous education in computer science. I had no experience in the field at all, which is why I wanted to take an introductory class (CS 1112 was full). There should be more classes geared towards actual beginners, as it is not reasonable to treat CS 1110 as a class that anyone can succeed in even without previous experience.
	You need more than one office hour. That's borderline pathetic given your paycheck.
	The POTDs could be way too hard and take way too long to complete three a week (I would sit down and work for 8+ hours...literally all day...trying to figure out some of the harder ones and still not get it).
	Too much work and time required for a 1000 level class
	interesting class, but the workload towards the end of the semester was excessive at times.
	I would just like to reiterate that the pace was too fast for an introductory course and the learning curve was too steep.
	one of hardest courses I've taken at uva. granted, I am a religious studies major.
	Class created a great base of knowledge about CS as I was previously not knowledgeable at all in the field
	Sometimes class went a little fast, but overall I thought it was a good class that was appropriately challenging and useful for the future. Reaffirmed my interest in CS!
	This course was very challenging, and I was surprised at the level of difficulty for an intro class that requires no prior programming experience. The students that had prior experience seemed to go through each assignment effortlessly, and the students without experience always seemed to be spending countless hours to figure difficult assignments out.
	Very effective
	TA's taught the course much too often. We pay a boatload of money to be taught by (supposedly) one of the premier professors in the field, not to have TA's struggle through class because the instructor is too busy to be bothered with showing up.

~ QUESTIONS AND DETAILS ~	~ ANSWER MATRICES ~
	I really enjoyed this course! I wish it was easier for people not in the E school to major/minor in CS/ wish there were more courses available to those not major/minoring in it. Tychonievich was a very enjoyable lecturer, he was great at explaining things but sometime came off as a little condescending when you would ask questions which is why I never came to his office hours. Overall, it was a great class and I learned a lot! Sometimes Tychonievich would get caught up trying to solve something complicated which would hold up the class a little, but other than that it was good! I am glad that not is a required E-School course. If it wasn't my perception of CS would be limited to the poorly taught high school class I took. I would likely not be majoring in CS. I am glad that the program is so fantastic here. Yay programming