

*** PROVISIONAL REPORT ***

UNIVERSITY OF TEXAS AT AUSTIN
 Downing, Glenn P C S371P 51670
 E100 EXPANDED

COURSE-INSTRUCTOR SURVEY
 OBJECT-ORIENTED PROGRAMMING

Fall 2018 DEPARTMENT COPY
 Enrollment = 60
 Surveys Returned = 57

	NUMBER CHOOSING EACH RESPONSE					NO. REPLIES THIS ITEM	AVG.
	Str Disag	Disagree	Neutral	Agree	Str Agree		
1 COURSE OBJECTIVES DEFINED-EXPLAINED	0	1	0	13	43	57	4.7
2 INSTRUCTOR PREPARED	0	0	0	12	45	57	4.8
3 COMMUNICATED INFORMATION EFFECTIVELY	0	0	3	11	42	56	4.7
4 STUDENTS ENCOURAGED-ACTIVE ROLE	0	0	1	15	41	57	4.7
5 INSTRUCTOR AVAILABILITY	0	5	4	18	30	57	4.3
6 COURSE WELL-ORGANIZED	0	0	5	12	40	57	4.6
7 STUDENT FREEDOM OF EXPRESSION	0	3	3	10	41	57	4.6
8 CLASS PARTICIPATION ENCOURAGED	0	0	0	6	51	57	4.9
9 ENGAGING INSTRUCTION	1	0	6	9	41	57	4.6
10 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT	0	0	0	6	51	57	4.9
11 INSTRUCTOR EXPLANATIONS CLEAR	0	0	2	11	44	57	4.7
12 GENUINELY INTERESTED IN TEACHING COURSE	1	0	0	8	48	57	4.8
13 HELPFUL COURSE MATERIALS	1	6	8	17	25	57	4.0
14 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS	0	1	5	15	36	57	4.5
15 ASSIGNMENTS AND TESTS RETURNED PROMPTLY	0	0	0	13	44	57	4.8
16 ASSIGNMENTS USUALLY WORTHWHILE	0	1	2	13	41	57	4.6
17 STUDENT PERFORMANCE EVALUATED FAIRLY	1	3	1	18	34	57	4.4
18 STUDENT PERCEPTION OF AMOUNT LEARNED	0	0	3	13	41	57	4.7
	Vry Unsat	Unsat	Satisfact	Very Good	Excellent		
19 OVERALL INSTRUCTOR RATING	0	0	4	9	44	57	4.7
20 OVERALL COURSE RATING	0	1	6	15	35	57	4.5
	Excessive	High	Right	Light	Insuff		
21 STUDENT RATING OF COURSE WORKLOAD	0	12	43	2	0	57	
	Less 2.00	2.00-2.49	2.50-2.99	3.00-3.49	3.50-4.00		
22 OVERALL UT GRADE POINT AVERAGE	0	0	4	14	39	57	
	<u>A</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>F</u>		
23 PROBABLE COURSE GRADE	32	18	5	2	0	57	

For the computation of averages, values were assigned on a 5-point scale so that the most favorable response was assigned a value of 5 and the least favorable response was assigned a value of 1.

COMMENTS:

Total Number of Comments: 37

-
1. The lectures were very clear and insightful. They're definitely some of the best I've had in UTCS. I feel like the projects were well balanced, so I wouldn't want anything changed about them. Quizzes could be a little more specific at times. Sometimes I got questions wrong because I misunderstood what they were asking for. Also, I wish the quizzes tested more important things. For example, I spent a couple hours diligently reading the textbook and learning the in-and-outs of OOP. However, the quiz was over how many neighbors a freckin cell has. No mention of any of the book reading. And I got the question wrong, despite studying for the material that will actually benefit me in the industry.
-
2. I thoroughly enjoyed the class. I thought it was great that we were encouraged to learn the various development tools (Docker, etc) via the projects in the class. I also thought the projects were quite fun, I probably enjoyed Darwin the most (cool results == cool assignment) I thought the cold calling in class was a good method of engagement. It keeps you on your toes and makes sure you pay attention rather than just letting yourself get lost in thought. Personally, I think it would've been nice to have one of the "learn C++" assignments exchanged for an object-oriented assignment. However, that may be because I was already comfortable with C memory management from CS439, people who haven't taken that class may have a different opinion.
-
3. Downing is very knowledgeable and knows so much about C++. He comes to every lecture so prepared and keeps class moving. Every lecture was engaging, and the format really encouraged discussion and thought over why the language works the way it does and how we should program in C++. The assignments were all easy enough that they didn't feel impossible but hard enough that they required some learning about new tools and some thought about object-oriented design and good programming practices. I really enjoyed this class and felt like I got a lot out of it (but you only get out what you put in).
-
4. TOOLS: all good; PROJECTS: - the goods: the number of projects & time given for each; grades returned promptly; structure & organization; interesting problems; class discussions relevant to project & explanation of specs with some examples; - the bads: more detailed written specification on HR for projects (sometimes there was unclear wording - which was usually explained in class, but it would be nice to have the HR for reference in case we miss something in lecture); only 2/5 of the projects really challenged us to think about object oriented design principles, it would be nice to replace either voting or collatz with a more OOP related project; SPEAKERS: didn't like that they were on friday nights
-
5. This course taught me a lot about C++ and introduced Object-Oriented concepts, just as I expected. I do wish that office hours weren't canceled as frequently with sometimes short notice; it got difficult to catch you for help at times. Otherwise, I really appreciated the way you taught the course with this Socratic methodology. It helps everyone stay focused and engaged during lecture, although it can be anxiety-inducing at times. If I had taken less CS courses this semester, I would probably have been able to put forth more effort to do better in this course. Anyway, thanks for the engaging lessons! Although I won't do the best in this course, I thoroughly enjoyed your passion and enthusiasm for the topics that you taught!
-
6. This class used HackerRank Tests, On my first test I was a bit unprepared for that kind of test which I haven't done before. I think a little bit more partial credit on the test would polarize the test grades a little less. The format of the test was good and forces you to practice coding and syntax but if you forget a little bit of syntax, you are in big trouble.
-
7. Downing has communicated the important aspects of Object-Oriented Programming through a clear understanding of C++ in such a manner that is understandable by most if not all. If something is not clear, Downing will answer any question in detail, whether in person or on Piazza. The material provided directly correlates with what he teaches in class, so paying attention in lecture is definitely a boon. And staying engaged in class is no problem, as much of the material covered in class is very hands-on.
-
8. The only problem I had with the course was that sometimes I felt we didn't learn everything that was necessary to complete the projects in class, so we had to figure a lot out on our own. For C++ issues that was fine with me, but it was sometimes frustrating trying to figure out how tools like Docker and govc worked on our own since we were just given the working makefile in the first project. Also, some of the C++ class material that was relevant to projects wouldn't be learned until just a few days before the due date. It would be more helpful if this material could be scheduled earlier or the projects could be scheduled later, so we had that knowledge going in.
-
9. I think this course was great, but I wish there was more of a focus on actual Object-Oriented Programming. This course was very much a course on C++ and less on OOP. Although the book was very OOP focused, the class was not and I think doing more OOP related work earlier on would have been nice. For example, the last two projects, Darwin and Life, were difficult but worthwhile because they forced me to use OOP principles. While I understand C++ requires a lot of teaching because it's a complex language, more time should be spent in class covering OOP principles and concepts.
-
10. Overall, I've learned a great deal in this course and have thoroughly enjoyed it. The tools we've been using are pretty helpful, but can also be tricky at times. In particular, Docker and Valgrind have both given me trouble and I would've liked if more help/resources were available for those. The projects have all been pretty interesting and definitely do-able. However, I don't like how Life was assigned right before break and due right after break - that made it a lot harder to work on. I also wish that the TAs answered more questions on Piazza, although I understand that maybe Professor Downing wants the students to collaborate more and attend office hours in person.
-
11. Very well organized course with a fair amount of workload.
-
12. I would have liked it if we could submit the projects earlier than just the 24 hours before so that when we are finished we don't have to worry about it anymore. I think that more people would post questions on piazza if it were anonymous to classmates. It would also help if office hours weren't canceled so often.
-
13. Although I have had a difficult semester with a heavy set of activities and extracurriculars, I really think this course is worthwhile
-
14. Liked switching from Github to Gitlab. Weekly readings - interesting and useful. Learning about Docker, Google test was really useful. Makefile - made workflow things like astyle and Doxygen are really easy. Travis CI was mentioned but I never needed to use it because of Gitlab so the time spent explaining Travis CI seemed wasted. I still don't understand why we needed to use cppcheck.
-
15. Officer hours were very flakey and prone to being cancelled, especially near assignment due dates. Piazza was completely useless for this class with instructors being very unhelpful and inefficient at answering questions on it. Other than these issues, it was a great course, I enjoyed the lecture format and the projects and exams.
-
16. As with most courses, I believe the tests are weighed too heavily. A reduction from 40 to 30-20 percent would change my

"evaluated fairly" answer to a "strongly agree". I don't believe these tests demonstrate how much I've learned in the course. Rather it's the projects, daily quizzes, and personal blogs that show-case this.

17. The tools we used weren't too much of a hassle to setup. However, I absolutely detested having projects and tests on HackerRank. Not being able to see the test cases or debug on HR was immensely frustrating. I understand it was to imitate industry, but even in industry there are ways to log your data so that it's not completely unknown. The quizzes did make me learn but were also unnecessarily stressful since they still affect my grade. Office hours were cancelled on a regular basis which was annoying when projects were due. Piazza was also not answered by the professor or TAs regularly and in a timely matter.

18. A wonderful class that made me realize how amazing object-oriented programming is. Great professor that makes you enjoy what you are learning.

19. I found the in class assignments and lectures to be extremely helpful, and the Projects were interesting, if a bit disparate to the topics we were currently learning in class. My main issue was that, compared to most other classes, this class was especially inaccessible to me, as a person who struggles with anxiety issues. I found the socratic method of lecturing and random calling to be fairly stress inducing for obvious reasons, and while I understand the benefit of writing out our experiences in class, making our responses public for all of the class to see was also harder for me. Thank you for making the whole "work with a different partner every time" thing optional for the projects, if it were mandatory this review would be different.

20. I really enjoyed the course, and I found the talks by external speakers very useful in thinking about my future career path. The projects were challenging but reasonable, however it would be nice if some of the material(e.g possible obstacles with using certain data structures) that is related to the implementation of the project can be covered before the project is released or within the first couple of days of the project released. Otherwise, it is a really great course and it has taught me a lot of C++ as well good object oriented design principles.

21. It was a little frustrating that you can't get any partial credit if you don't pass the HackerRank tests. It was unfortunate not having anyone who held office hours in the middle of the week because I had a problem where I needed someone to look at my code in order to solve it the day after a project was due, but no one was available.

22. Great class, but the first couple of projects didn't feel like they specifically were tied to OOP.

23. You are really funny and engaging. I think that helps us all learn a little bit more.

24. I feel like both OOP and SWE lose something by being so similar. For example, GitLab pipelines make a lot of sense to do in SWE: you can set it up to run the test suite and then deploy on pass. It can also block incorrect code from being merged intomaster. For OOP, it seemed like more of a chore. All it did for us was run tests on master, telling us what we already knew from running it locally. Additionally, we only had groups of 2, so there was no need to dev on other branches or anything. There are several other instances where it feels like you're keeping the courses the same for the sake of keeping them the same, rather than it being the best thing for either course. Just something to think about.

25. The course was great, but the workload was pretty high and we really could've used some slip days or something on the projects. If you miss the due date by one day then its suddenly 20 points off and it would be nice to at least have 1 or 2 slip days total.

26. Sometimes you talk over people's answers which can be offputting.

27. I thought the projects we had and all the development tools we had to use were really good. I'm usually not an organized coder so I actually had to think about design and style in a class and not just if the tests passes. I didn't like Docker thoughsince I use a Windows machine and had to switch back and forth between a vm and my local machine to complete projects which was tedious. I also did not like the test style since I tend to make a lot of coding errors especially on a time limit.

28. Amazing course and teaching style.

29. SUGGESTIONS: -Write out the main points of each lecture in the notes (sometimes it's hard to determine the core lessons from looking at a block of code, and the core lesson may not always be recorded in a student's notes) -Reduce the readings, especially if they are rarely evaluated -Spend more time teaching real world skills like CI pipelines, makefiles, issue tracking, UML diagrams, unit testing etc. Some of these concepts were hard to learn on my own. -BUT don't make us do all that stuff foreach project. Maybe do one project where you have to carefully track issues, and one project where you have to make a really good UML diagram. But doing some of these chores for every project gets tedious.

30. I really enjoyed your class. I don't think it was so much of an object oriented class. Rather it felt like a class on C++ and a class on how important it is to know how a langue is compiled.

31. Lectures were really good an engaging, projects were fair and interesting. C++ is massive but more detail into smart pointers and lambdas would have been nice.

32. I enjoyed this class and learned a lot! One tool that I liked using for the projects was GitLab, which I had never used before. It was so useful for continuous integration that I decided to use it for another class this semester. I appreciated that there was never a project due the week of the exams in this class. Thanks!

33. Great professor.Sometimes I feel like we didn't learn important information about the projects until after we finished them, but maybe that's how the course was designed.

34. Downing is a great professor who always brings great energy to class and communicates information clearly and effectively. This class is definitely one of the better electives available since you leave with knowledge that can be applied to a CS career. The projects were fair in workload, but working with the extra requirements of gitlab, makefile, tests, etc, can sometimes be time-consuming. I personally always had issues with the makefile and things with docker that slowed down my progress, although I already had the solution working in hackerank. Some more helpful guidelines with extra requirements could be useful. But overall, I really enjoyed this class and I would love to take another class from Professor Downing.

35. At first, I thought this class was going to be a lot of work for very little gain. In the end, it ended up being a lot of work but I would not say that the rewards reaped were not substantial. I learned a great deal about c++, architecture and design from just this one semester alone and I am truly grateful to have been given the opportunity to take this class. I enjoyed using all of the tools besides Docker but that's because it's kind of complicated setting it up. The projects were all worthwhile but I did not like the fact that Life was assigned over the break. It's pretty difficult to pair program when you and your partner are miles away from one another. The speakers from Atlassian and Under Armour were great. Overall, I enjoyed class.

36. This class was an amazing class. Every lecture felt very organized, and felt like I was learning something new. I feel like your teaching styte is very effective

37. Maybe spend more time explaining how to properly manipulate the stack to avoid using the new keyword in Darwin and Life, I probably spent the vast majority of the time in those assignments trying to figure out how to do that and debug issues with it that didn't work
