*** PROVISIONAL REPORT ***

UNIVERSITY OF TEXAS AT AUSTIN — COURSE-INSTRUCTOR SURVEY — Spring 2016 DEPARTMENT COPY
Downing, Glenn P — C S371P — 51210 — OBJECT-ORIENTED PROGRAMMING — Enrollment = 65
E100 EXPANDED — 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	0	1	15	41	57	4.7
2 INSTRUCTOR PREPARED	0	0	0	2	55	57	5.0
3 COMMUNICATED INFORMATION EFFECTIVELY	0	0	1	7	49	57	4.8
4 STUDENTS ENCOURAGED-ACTIVE ROLE	0	0	6	9	41	56	4.6
5 INSTRUCTOR AVAILABILITY	0	0	4	21	32	57	4.5
6 COURSE WELL-ORGANIZED	0	1	1	6	49	57	4.8
7 STUDENT FREEDOM OF EXPRESSION	0	0	4	15	38	57	4.6
8 CLASS PARTICIPATION ENCOURAGED	0	0	2	6	49	57	4.8
9 ENGAGING INSTRUCTION	0	0	1	12	44	57	4.8
10 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT	0	0	0	7	50	57	4.9
11 INSTRUCTOR EXPLANATIONS CLEAR	0	0	1	12	44	57	4.8
12 GENUINELY INTERESTED IN TEACHING COURSE	0	0	1	7	49	57	4.8
13 HELPFUL COURSE MATERIALS	0	7	8	15	27	57	4.1
14 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS	0	1	8	14	34	57	4.4
15 ASSIGNMENTS AND TESTS RETURNED PROMPTLY	1	0	3	21	32	57	4.5
16 ASSIGNMENTS USUALLY WORTHWHILE	0	1	4	16	36	57	4.5
17 STUDENT PERFORMANCE EVALUATED FAIRLY	0	4	7	25	21	57	4.1
18 STUDENT PERCEPTION OF AMOUNT LEARNED	0	3	2	13	38	56	4.5
	Vry Unsat	Unsat	Satisfact	Very Good	Excellent		
19 OVERALL INSTRUCTOR RATING	0	1	2	8	46	57	4.7
20 OVERALL COURSE RATING	0	3	5	11	38	57	4.5
	Excessive	High	Right	Light	Insuff		
21 STUDENT RATING OF COURSE WORKLOAD	2	18	35	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	1	1	6	23	25	56	
	__A__	__B__	__C__	__D__	__F		
23 PROBABLE COURSE GRADE	15	27	13	1	0	56	

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: 43

---

1. One of my favorite electives I have taken so far in the CS. I knew going into the course that the focus was more on C than OOP design patterns, so nothing was unexpected. I enjoyed everything about the course and would enthusiastically recommend it and Professor Downing to anyone.

---

2. This class is a lot of fun. The projects start easy and gradually get harder. I really enjoyed the Allocator project, because it taught me how the heap works and reinforced my knowledge in pointers. The lectures are really good and engaging, I have never once fell asleep in his lectures. Finally, I learned more about Git such as closing issues with commits and resolving merge conflicts . Thanks Downing for making us practice good coding styles and using Github!

---

3. I think the projects were weighted too lightly and the tests too heavily. I spent the vast majority of my time for this course working through those projects, and I learned a lot from them, but they're worth so much less than the exams. The grading does not accurately represent the time and effort put in. The exams are necessary, but it doesn't make sense that a class that specifically tries to prepare us to work in industry would place so much emphasis on a construct that doesn't exist there.

---

4. I thought the course was worthwhile and valuable. One complaint I have is the grading scheme is heavily skewed towards quizzes. I would humbly recommend reducing the impact of quizzes, as I feel it causes more stress than a way to bring people to class should cause.

---

5. I actually enjoyed the quizzes and randomly being called on because it encouraged me to stay on my toes and constantly keep up to date with the material. That was especially helpful when it came to studying for the exams because I had been studying all along to perform well on the quizzes. The projects were usually very straightforward but I found the amount of unit tests we were required to right to be extremely tedious compared to how simple straightforward the code was. In all, I learned a great deal in this class and wish I would have taken in early in college career.

---

6. Overall I felt the course was great and I enjoyed your lectures. The projects were fair I thought. I did feel the turn-in requirements for the projects were a bit excessive and added a lot more unnecessary stress to the assignments. I think in the future, students need more preparation for the Life assignments. I worked with a partner on this assignment, so we were able to figure it out before some of the later lectures. If I had worked by myself though, I think I would of been in a lot more trouble. I also felt the pass code for the quizzes was a little too nit picky as well. I enjoyed the practical nature of your course and think more courses taught at this school need to be like this. Guess lecturers were great btw. Thanks.

---

7. The course was very well organized. I really appreciated the detailed knowledge taught about C plus plus and Java. Knowing things work is one thing, but knowing how they work is priceless. Hence, really appreciated the effort. I was very hesitant with some of the tools before this class, but along the course I have become more comfortable with them and started appreciating their existence for instance github and GTest. Out of the speakers the two which I thoroughly enjoyed and who were the mostinformative for me were from Main Street Hub and Atlassian. The speakers were very thorough and talked about experiences that I could relate to and tools that I would like to incorporate in my work life.

---

8. I think the projects should be worth more of the overall score for the class since we spent so much time on them each week, rather than the tests being worth most of the grade.

---

9. I enjoyed this course a lot and I learned a lot from class and from the projects. It would have been great if I have knew about in my junior year because this class definitely would help students prepare for interviews and get internships. The only problem I have with the course is taking exams on Canvas. Even though it is more environmental friendly, it was really stressful taking the exam online, especially the way questions are formatted on Canvas is just really annoying. I feel like I wasted some time scrolling around in between the instruction and the coding text box, which could have been avoid if there is a way to see them side by side. But rest of the class is great this class is the most educational and informative elective in CS

---

10. Didn't like the in-class quizzes. Sometimes had technical issues with phone or internet connection preventing me from taking the quiz, or taking up time that could have been used for answering. Feel like they take up time that could be better usedlecturing.

---

11. I really like the class as overall. I understood C , pointers, and implementation underlying certain data structures better. Though I feel like Life and Darwin are similar, they help me understand different concepts. I know it is a part of learningexperiences in your class to get called. It helps me stay focus and alert. I think it might be slightly better if your roll calls are more randomized. Though I really like github and issue tracker, I wonder about BitBucket and Jira since several speakers have mentioned them. I think Jira will be a really cool tool to use, especially in a team environment.

---

12. I learned a ton in this class!

---

13. Awesome class. Professor Downing was by far the best instructor I have had for any class at UT. If every one of my professors was like him, my GPA wouldn't be as bad as it is. Whenever someone had a question, he wouldn't just answer it but he would help guide them to figuring it out themselves and why it was so. The office hours were great too, he answered all my questions about class and my personal projects. Thank you so much for teaching this class. I wish all my professors were like you.

---

14. I liked having all of the quizzes on canvas, but I wasn't as happy with the exam online. Excellent lectures, but the supplemental material wasn't very good. The textbook was expensive, rarely used, and not very helpful.

---

15. I thoroughly enjoyed this class. Professor Downing was always excellently prepared to explain the concepts we went over in class. The projects outside of class were a bit frustrating and I feel they would be less so if all implementation details minus interfaces were left to the students. Or if specific implementation details were always mentioned in the assignment, rather than tacked on later.

---

16. I enjoyed the lecturing style and Google Doc, and walking through exactly what was happening in a line of code was particularly helpful. It would be nice if the project specs were more detailed to avoid clarifications near the deadline that requiredcode rewrites. Also, the midterm didn't seem to focus on what was emphasized in class. Overall I learned a lot from this class, especially about what goes on "underneath the covers," but I felt the course was misnamed since we spent so little time on OOP

concepts.

---

17. Git and Github are always right. Gtest was great because it taught me unit testing. Didn't really use gprof, but gcov was interesting to look at. Glad you had us using Valgrind. Didn't really look at the Travis file since it was already working for us, but that's on me. Doxygen was new and very useful, glad you included it. Each project taught me more about C or the workflow, so all of them were useful. Life taught me the most about OOP principles, but you might want to show us the Handle class earlier if that's how you want us to implement Cell. That way we can just inherit from it. Atlassian was the best for learning the workflow. Mutual mobile was interesting to learn about open source tools. Missed the others unfortunately

---

18. I thoroughly enjoyed the class, however found that the quizzes were weighted a bit heavy for what they were. The time was typically too short and forced me to take a guess. Also being able to go back on problems would have been very beneficial.

---

19. This was one of my favorite classes here at UT. I liked the tools that we used for the class, but some instructions on how to set them up on your personal machine would be a huge help, I never did get them set up at home but a tool to suggest to users is sublime as it can connect to ssh and it's super simple to set up . I loved the projects and every day in class was just as great. The in class quizzes were fair, but sometimes the 3 minute time limit was not enough, and some times it felt as if the questions were just trying to trick us near the beginning of the semester more so . The speakers were not that intriguing. They got repetitive after a while and I never really learned anything from them I have worked in the field 2 years tho

---

20. I very much enjoyed Professor Downing's lectures and he has definitely set a new, higher bar on what I would expect out of a lecture. Classes were engaging, he was patient and precise when answering questions, and he had helpful words for situationsthat did not directly pertain to his lecture's content, like career advice. I would recommend improved project pages. For example, Allocator's specification section had scarce info, and it would have been impossible to get started without asking the professor for a full explanation on the assignment, since I had missed lecture that day. I believe that should not be the case, since others should be able to get important info from the webpage at any time, independent from previous events.

---

21. 1. If you have used the tools before, the class is not very useful in terms of tools at least. What would be cool is to do Code Reviews. THAT WOULD BE REALLY USEFUL. Learn how to get and receive feedback from colleagues. CPP - I don't like that you teach PL specifics in class - just give us short weekly assignments to learn about inheritance, pass-by-reference and so on. We would remember it better and wouldn't waste class time. 2. I wish projects were more complex and big. 3. Some speakers were boring especially Rackspace because they didn't talk about specific code. !!!This class should be taught alongside CS 314. Would be useful before people had their first internships and haven't used tools in class!!!

---

22. I understand that your course relies on class participation, however, I don't think students should be penalized for missing quizzes due to interviews and or illness. I think that you should remove the daily code for the quizzes. Even though you drop quizzes I feel this would be more fair. You may have tried this in previous semesters and it did not work, however, this is my opinion. I never attended your office hours, however, when I approached you after class with a question I felt that you were a bit dismissive. This made me hesitant to ask questions outside of class time in which you were very willing to explain things and answer questions . I really enjoyed the course and am looking forward to your SWE course next semester! Thanks.

---

23. Up until the last few weeks of class, it seemed like there was little talk of object oriented design during lecture. All of the speakers were great. The quizzes can be a little disheartening at times. I think it would be nice if simply finishing the quiz got you a point. But I understand the reasons against this.

---

24. I really enjoyed the presentations from local companies here in Austin! It's great that we can learn practical stuff like that in our courses at college. I liked that this course had a strong emphasis on learning technologies that we will later use in our post-college career. Courses like this make my investment at UT feel worthwhile.

---

25. Great class. Tough but I learned a lot.

---

26. I like the old classroom in CLA better, a dark classroom with code on screen makes me fall asleep and hard to takes notes. The material covered is excellent, right on point. One thing I'd like to see is a clearer project instruction, I know it has flaw and we need to improve it, but sometimes I find it irritating to not able to start on the first day after project is out. Maybe a slightly clearer instructions set? Canvas works very well, far better than iclicker and tophat, the only problem will be the quiz locked issue, but that's Canvas issue. Overall, the course is excellent and I benefit a lot from it.

---

27. Out of the 5 projects, Voting didn't seem necessary. Also, I would much rather have preferred to take time creating the makefiles and .travis files for each project rather than have them handed to us. Also it would've been nice to take time to talk about making tests for many than a day or two. I loved using travis and GitHub but would've preferred using bitbucket because of the private repos. Using a student account gives you 5 private repos and there are 5 total projects. This means you must delete repos or make them public.

---

28. Please use bitbucket instead so I don't have to give up all my private repos that I was already using. All projects are way too easy, and I didn't get much out of them. The strict requirement of having 3 unit tests per method encouraged bad design.Lump all the functions together, and the student is rewarded by having to write less unit tests. Good design is asked for, but the requirements discourage good design. I've had multiple partners that were not interested in good design. Use code coverage as the proper metric if you want students to actually test their code. The class itself is extremely slow and needs to move a lot faster. Let students be responsible for learning the material. Also, midterm was way too easy.

---

29. My one main concern with the course was with project requirements. Often times some files were incorrectly given i.e. .travis.yml or other requirements modified or added i.e. include constant begin and end methods , which were not outlined orupdated in the project page. Instead, these updates were posted in Piazza. It would have made the experience of doing the project more straightforward if files and other requirements were added to the the project guideline page as well as being posted on Piazza.

---

30. The main objection I had with the class was that the title "Object Oriented Programming" was a bit misleading. This was somewhat evident in the syllabus, but I didn't think much of it. Sure, we read papers about proper OO design, but I wished we would've had more discussion about the topics we read. Even better, I hoped we could apply these practices to more than the 2 projects we had. This is what I imagined learning object oriented programming would have been like. However, we did get into good OO design later in the semester, but I wished we had spent more time discussing OO design. That said, I did learn a lot of C and the details about C that would've been really difficult to learn on my own.

---

31. Professor Downing is awesome. He is a very animated lecturer who really knows his stuff. I thought all the tools we learned GTest, Travis, git, ect. were generally pretty useful, especially as someone who had only really used git before. The projects were a little easy in terms of actually developing the algorithms, but as a vehicle to teach concepts they worked pretty well. With the guest lecturers, I thought that they were pretty good for the most part. However, they generally felt a littleunfocused. Obviously it's tough to grade the quality of a lecture when you're asking someone to give up their time to come in, so I will say that I did learn something from every guest lecturer.

---

32. I felt that the course definitely gave great insight and teachings on the topics of Object Oriented Programming. I was very unconfident before, but now I feel thoroughly satisfied and confident with the material. Pros - quizzes are good - callingout in class is great - projects are perfect - topics and pace is great Cons - Canvas test - I understand its to make a paper free environment, but the fact that the tests aren't well formatted for code segments makes it time consuming and wasteful. I totally understand that it speeds up the grading though.

---

33. I thought this was a very good course overall, and I felt I learned a great deal about how to develop better-designed and more maintainable software from it. I might also mention this course felt like one of the best designed and executed I have ever had. The only real complaint that comes to me off the top of my head is that I felt some of the questions on the quizzes tested more whether we had done the work reading than whether we had actually learned anything for example, asking where one of the writer worked rather than about what insignts we took away from a particular article . I do not find quizzes of this design to be as meaningful as those that focus on material which it would actually be useful to learn.

---

34. One of the best professors in the CS department. He gives very clear definitions of what he expects from the students in his class, and his lectures are engaging and interesting. He's funny, witty, and doesn't get mad when you notice that he makes asmall mistake. I love the way he keeps track of all the course material on a calendar and how he writes down class notes in a google doc. Overall excellent professor and I can't wait to take another class with him.

---

35. I enjoyed gaining experience in using git and Travis CI, but I wish I understood the numbers from gcov and gprof better. They're still a little mysterious about what they mean and how they work. I hope that usage of appropriate getters and setters will be clearer in the future as well.

---

36. I felt that the class didn't go through Object Oriented Design as much as it should have. It was very much a c class and for the most part didnt learn anything brand new, but refreshed myself on c and all the nuances in it. I think it would benefit people if you focused more on OOP.

---

37. Hands down one of the best courses I've taken at UT. I love the way you prompt, explain, and connect concepts in lectures. One thing that could be improved is eliminate instances when we scroll everywhere in different files talking about different pieces of code. It gets confusing even when the concepts themselves aren't hard. The projects themselves are good, I think you could leave more implementation algorithm solutions for us to solve though. I personally didn't benefit from the speakers atall, probably because I've seen a lot of this kind of talks. Quizzes are more stressful than helpful Internet connection time pressure . Lastly, thank you so much for CARING about teaching us. You are one of the best professors there is.

---

38. It's sad that I might be 1-2 pts off an A and still be evaluated equally to those who performed a lot worse than me. But I didn't take this class to get a grade. I was always amazed at how fast he was able to answer questions, many times with counterexamples. It just shows how thoroughly he knows the class material. He also lectures in an organized, uniform manner. To further my point and show Downing's extremity when it comes to uniform lecture style, he uses catch phrases such as "crystal clear", "appreciate the fact", "doncha think". This unformity makes you feel comfortable about what to expect and so you're able to digest information better.

---

39. Excellent course, though more resources would have been greatly appreciated. The entire course moves at a breakneck pace, which was mostly a positive but I felt like a lot got lost in the cracks. The tests were very punishing, especially because it felt like you had to know the tested material perfectly to do well, yet it was often unclear what the test would be over. For example, while we covered some concepts that the book also covered, I don't think it was ever mentioned in class, and it was always unclear which areas we should be focusing on. A practice test would be very helpful in this area. Downing is a fantastic professor who I would and have recommended to pretty much everybody.

---

40. Quizzes are rough....demoralizing.... Maybe assignments should be worth more and tests a bit less, maybe 40 each.

---

41. Downing was very funny and I really enjoyed the course. I think all of the tools we used in doing the projects were satisfactory but I would have liked to have more information given about Google Test, specifically the friend tests. It was a bit troublesome to figure that part out so it would have been nice to have had an explanation of it. The projects were well designed to go over what we were learning in the class at the time and I also thought the speakers were a great way expose us to companies and real-world applications of what we were doing. They also motivated me to look for more job and internship opportunities.

---

42. I am always interested in different ways of teaching and learning and I really enjoyed Prof. Downing's novel approach to teaching. It had high engagement and reinforced concepts verbally, visually, through tests, and through practice. I actually really enjoyed learning how code works through the informative tests. I enjoyed the topics covered in class and feel that they were relevant and useful. Taking tests on Canvas was a pain and the tests felt very rushed and very short on time. Projectswere very good learning tools, but I wish UML was covered in class. I would appreciate if we could go back to previous questions on the daily quizzes. The talks were pretty interesting, but a bit too similar. I did enjoy the open source talk.

---

43. The material I learned in this class was vital to acquiring my summer internship. Though my performance was rough due to some personal incidents, Downing is one of the best teachers at UT and helped me go from someone who knows nothing about C to being an expert at it in one semester.
