

*** PROVISIONAL REPORT ***

UNIVERSITY OF TEXAS AT AUSTIN
Downing, Glenn P C S371P 53525
E000 Expanded

COURSE-INSTRUCTOR SURVEY
OBJECT-ORIENTED PROGRAMMING

Spring 2013 DEPARTMENT COPY
Enrollment = 61
Surveys Returned = 42

	NUMBER CHOOSING EACH RESPONSE					NO. REPLIES THIS ITEM	AVG.
	Str	Disag	Disagree	Neutral	Agree	Str Agree	
1 COURSE WELL-ORGANIZED	0		1	0	3	38	4.9
2 COMMUNICATED INFORMATION EFFECTIVELY	0		0	1	8	33	4.8
3 SHOWED INTEREST IN STUDENT PROGRESS	0		1	2	15	24	4.5
4 ASSIGNMENTS AND TESTS RETURNED PROMPTLY	0		0	4	15	23	4.5
5 STUDENT FREEDOM OF EXPRESSION	0		3	1	12	26	4.5
6 OBJECTIVES/ASSIGNMENTS CLEARLY STATED	0		2	0	10	30	4.6
7 INSTRUCTOR WELL-PREPARED	0		0	0	1	41	5.0
8 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT	0		0	0	3	39	4.9
9 GENUINELY INTERESTED IN TEACHING COURSE	0		0	0	4	37	4.9
10 AVAILABILITY OUTSIDE OF CLASS	0		0	7	19	16	4.2
11 STUDENT PERFORMANCE EVALUATED FAIRLY	0		2	7	21	12	4.0
12 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS	0		0	2	16	24	4.5
13 COURSE WAS MADE EDUCATIONALLY VALUABLE	0		0	0	7	35	4.8
14 INSTRUCTOR INCREASED STUDENT KNOWLEDGE	0		0	1	7	34	4.8
15 INTELLECTUALLY STIMULATING	0		1	1	13	27	4.6
16 ASSIGNMENTS USUALLY WORTHWHILE	0		0	0	14	28	4.7
17 COURSE OF VALUE TO DATE	0		0	0	10	32	4.8
	Vry	Unsat	Unsat	Satisfact	Very Good	Excellent	
18 OVERALL INSTRUCTOR RATING	0		0	2	8	32	4.7
19 OVERALL COURSE RATING	0		0	3	12	27	4.6
	Excessive	High	Average	Light	Insuffic		
20 STUDENT RATING OF COURSE WORKLOAD	3	12	25	2	0	42	
	Less	2.00	2.00-2.49	2.50-2.99	3.00-3.49	3.50-4.00	
21 OVERALL UT GRADE POINT AVERAGE	0		5	5	16	16	42
	A	B	C	D	F		
22 PROBABLE COURSE GRADE	5	26	9	2	0	42	

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: 24

1. Excellent course...really glad I took it.

2. I loved the course. It was a fantastic way to expand my C knowledge and overall programming experience. The only thing I would recommend changing is try and make it more obvious what the coding questions could be over on the exams. I had a good idea for exam 1, but after that I had no clue.

3. I very much enjoyed the course. I have learned a lot from Professor Downing and he was a great instructor. However, I feel that my grade in the course did not at all represent the work and effort I put in. The projects and quizzes accurately reflected my grade in the class. The tests did not pan out so well. It felt like they just involved too much information and required an exact and precise understanding of everything. While I understand that it's important to know the fine details, I don't think that is the material that should be tested. Concepts and broader ideas should be tested and maybe leave the specifics for in-class quizzes or possibly homework problems.

4. Professor Downing's courses are always incredibly educational and go into depths that other classes do not. The assignments, quizzes and exams are also great learning experiences, but the grading scheme is a little harsh for the level of difficulty. It would be nice to have more practice questions that would not only prepare us better for tests, but also enhance our understanding by complimenting what is taught in class.

5. Great course. Tests are difficult, but I like how well the material lines up with the projects

6. Looking forward to Software Engineering with Downing

7. I think the course should be considered more of a C course, not an "OOP" one. Letting students choose between C, Java, and Python for each project, and allowing them to use that language twice during the course of the year would work. Tests are just plain too tricky. There's hardly a correlation of what we talk about in lectures to what's on the test. If your multiple choice average is 65-70 I think that's too low. Projects should also not be all or nothing type projects like they are. You could literally do nothing and get the same grade pulling all nighters and it just doesn't pass Sphere. Shit happens - students shouldn't be given zeros for trying. Other than that it's a great course, and it makes you work hard and learn a lot.

8. Great course, though the grading was kind of silly. The exams didn't seem to reflect what I'd learned very well, and getting nickled and dimed on projects was incredibly frustrating when the core concept was clearly demonstrated.

9. This was without a doubt the best course I took in my four years here. Thank you for being an excellent teacher!

10. I thoroughly enjoyed the class and both will recommend and have recommended this class to others for the future. I do feel that the tests were extremely difficult and that our class's grades overall will be very low in the case that the third exam is as difficult as the second. I suppose by the time you read this that will already have occurred, but in my position with a Z-score of .7 that puts me around the edge of the top 25 of the class and with me being unlikely to get an A assuming I get an 80 on the Life project, I will need a 150 on the that seems a little bit low for the class.

11. Professor Downing definitely lived up to and far exceeded my expectations. Every CS student at UT knows that they need to take a Downing course once in their life. Even though this was a C class, it further expanded my knowledge of the inner workings of Java. This class experience is definitely one of the high points of my educational career. Keep it up Downing! In an age where most students are concerned about their GPA, you sure make learning the material fun and interesting.

12. I felt like the tests were much more difficult than they needed to be.

13. I felt like there was a disconnect between the readings and the lectures. The readings were more focused on OOD and the lectures were more focused on syntax rules and details. Also, the lectures were far too fast-paced for me with an overwhelming amount of details to memorize within a given time period, many times I could not keep up and missed a lot of important parts of the lecture. Regardless, I did learn a lot about the C language.

14. I LOVE DOWNING best CS professor I have had. Can't think of anyone that even comes close to him.

15. I loved this class. Such a great way to learn C and the STL. Professor Downing is an amazing professor and his ability to communicate the subtleties of the C language was amazing. My only quarrel would be that this class was more of "Advanced C Programming" than "Object-Oriented Programming". While learning the C language is invaluable, I felt like there could've been much more emphasis on actually applying OOP concepts. The programming projects were at the level of those I did in my high school computer science classes. but in C It would be nice to more complexity and possibly large projects in large groups that would force the use of OOP rather than just conforming to a spec. Looking forward to Software Engineering!! D

16. Downing really knows his stuff. This was a very intense class but I think it'll be very useful. His teaching method forces you to pay attention, which is good. In class, he makes you feel good when you get a question right, but sometimes he made students feel dumb for saying the wrong answer. I found the tests to be extremely tricky, especially the 2nd one. A lot of the multiple choice questions had 2 parts to them, but you could only choose 1 answer. On all but 1 of those questions that I got wrong, I had half of the answer right only got half wrong. So if those questions had been split up 1 part per question I would've gotten more points. I also didn't have enough time to finish the tests. Overall, challenging but useful. Thank you!

17. Downing is an awesome professor.

18. Overall very good, interesting lectures. Kept us engaged, though sometimes he could be a bit unpleasant when a student could not give the correct answer to the question he asked. An example is when he asked the whole class to give incorrect answers in unanimity. Also, I felt the exams were designed to trick us instead of assessing our knowledge.

19. Exams seemed a bit tricky. While the content was definitely fair, I was caught off guard by exactly how specific the exams would be. Most CS courses at this level are about conceptual learning with the idea that syntax can be reviewed on the job with a reference. Clearly, syntax is important in this class, but I was caught unaware. Otherwise, I loved the class. The programming parts of the exams were fun perhaps the only time I've said that about an exam and the projects ramped well to get more and more involved as we learned more. Thank you for making a course so useful for my career.

20. I really liked the teaching style professor Downing used in class. In particular, addressing students individually while teaching. It keeps us on our toes because we need to be prepared when called on, and it keeps class moving because we aren't waiting for

one person to volunteer an answer. I found it engaging and effective. Sometimes the tests seemed to cover obscure, unnecessary details of the subject, but that is my only complaint. In general, i really enjoyed the class!

21. Very hard class, but definitely worthwhile to take. I do wish that we had homework assignments, but you can't have everything. This class was supremely useful and Dr. Downing is an outstanding professor. The only other thing that I would wish for is a curve, at least on the exams.

22. Multiple choice questions on tests aren't really fair in a major as open ended as ours, and as a result I think a lot of people's test scores don't really represent how much people learned. Other than that, great class!

23. Excellent course! Your style of lecturing is very good for this material and I'd like to see you explore teaching other courses esp. algorithms if you are comfortable. I'd like to see you use standard file naming conventions -- e.g., file.cpp instead of file.c, and for headers use file.hpp instead of file.h since they are Cplusplus headers not C headers. Also, early on, the time for quizzes was insufficient occasionally but I believe you found a good balance during the last few weeks.

24. On the class website, the summary of the amount of time people should expect to spend on this class needs a note next to it saying that this is IF you are already competent in C. If not, more time is required, especially with studying for the exams. I really enjoyed working on the projects. However, I would have liked to work on more projects like Darwin and Life instead of the first two projects we had. Darwin and Life were far more interesting. I didn't perform as well on the exams as I'd hoped. Initially I thought it was due to the course structure, but now I think it is entirely due not spending the time to play with the example code to truly understand the concepts. The study group next semester will be a good way to enforce this.
