```
                                                    *** PROVISIONAL REPORT ***

      UNIVERSITY OF TEXAS AT AUSTIN                   COURSE-INSTRUCTOR SURVEY           Spring 2017 DEPARTMENT COPY
      Downing, Glenn P          C S371P     52195     OBJECT-ORIENTED PROGRAMMING        Enrollment = 70
      E100 EXPANDED                                                                      Surveys Returned = 65

                                                                                                            NO. REPLIES
                                                                 NUMBER CHOOSING EACH RESPONSE               THIS ITEM     AVG.

                                                    Str Disag  Disagree    Neutral     Agree    Str Agree
     1 COURSE OBJECTIVES DEFINED-EXPLAINED               0          0          0         14         51           65        4.8
     2 INSTRUCTOR PREPARED                               0          0          0          4         61           65        4.9
     3 COMMUNICATED INFORMATION EFFECTIVELY              0          0          0         11         54           65        4.8
     4 STUDENTS ENCOURAGED-ACTIVE ROLE                   0          0          2         17         46           65        4.7
     5 INSTRUCTOR AVAILABILITY                           0          0          4         14         47           65        4.7
     6 COURSE WELL-ORGANIZED                             0          0          2         14         49           65        4.7
     7 STUDENT FREEDOM OF EXPRESSION                     0          1          4         19         41           65        4.5
     8 CLASS PARTICIPATION ENCOURAGED                    0          0          0          5         60           65        4.9
     9 ENGAGING INSTRUCTION                              0          1          2         16         46           65        4.6
    10 INST. HAD THOROUGH KNOWLEDGE OF SUBJECT           0          0          0          3         61           64        5.0
    11 INSTRUCTOR EXPLANATIONS CLEAR                     0          0          0         21         44           65        4.7
    12 GENUINELY INTERESTED IN TEACHING COURSE           0          0          0          7         58           65        4.9
    13 HELPFUL COURSE MATERIALS                          0          3         19         14         29           65        4.1
    14 ADEQUATE INSTRUCTIONS FOR ASSIGNMENTS             1          1          3         26         34           65        4.4
    15 ASSIGNMENTS AND TESTS RETURNED PROMPTLY           0          0          1         15         49           65        4.7
    16 ASSIGNMENTS USUALLY WORTHWHILE                    0          1          3         20         41           65        4.6
    17 STUDENT PERFORMANCE EVALUATED FAIRLY              1          1          5         20         38           65        4.4
    18 STUDENT PERCEPTION OF AMOUNT LEARNED              0          1          2         13         49           65        4.7

                                                    Vry Unsat    Unsat    Satisfact  Very Good  Excellent
    19 OVERALL INSTRUCTOR RATING                         0          0          0          9         56           65        4.9
    20 OVERALL COURSE RATING                             0          0          3         15         47           65        4.7

                                                    Excessive  High       Right      Light      Insuff
    21 STUDENT RATING OF COURSE WORKLOAD                 0         16         44          4          1           65

                                                    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          3         15         16         30           65

                                                      __A__      __B__      __C__      __D__      __F
    23 PROBABLE COURSE GRADE                            27         32          6          0          0           65
--------------------------------------------------------------------------------------------------------------------------------
                 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: 36

----

1. This class was the first time I've genuinely enjoyed a computer science course and actually felt like I had learned something worthwhile. Downing was a fantastic professor even if I always feared being called on in the class.

----

2. The daily quizzes can be a brutal hit to the overall course grade - even if the student makes it to every class. Please consider making them participation grades or giving a chunk of their weight towards projects.

----

3. I greatly appreciated learning how to use clang-check and format in this class. I thought that Docker was interesting but the instructions on using it weren't clear enough for me to use it. I found doxygen to be a very useful documentaton tool. Gcovhas a lot of usability but I think some time should be spent in class teaching students how to use the friend test. I already knew how to use git but I'm glad I was able to practice it more. Travis CI was by far the most useful thing I've learned about, and mentioning the use of CI in interviews was integral to getting my internship. I though all of the projects were interesting, but I think that Life should come before Darwin due to the difficulty difference. -- Out of space but GREAT CLASS!!!

----

4. Getting exposure to all the tools we used was an excellent intro. It forced me to at least familiarize myself with the basics. Docker seems like it is more powerful than I used it, but it was incredibly useful. It would be especially helpful in a graphics or gametech class. The projects were absolutely worthwhile. They built directly on concepts from class, while still requiring some thought.

----

5. First off I really enjoyed this class. I hate clang's format and it kills me every time I see it. Projects were all interesting and straight-forward. I wish Downing would go over what makes effective unit tests. None of the speakers were that interesting honestly. Definitely felt like listening to them was a waste of time and that I was just in it for the points. I actually go to a lot of talks hosted by the department so I'm just saying that the ones we listened to as a class were not that great. Please downing go over your quizzes before class and check for typos. The cost of using the Moore method of teaching is that we don't get through material as fast as I would like. I feel we could have learned a lot more in class than we did.

----

6. Really great class and really great professor. The lectures were engaging and the information was conveyed well. The projects were useful and not too difficult. The tools we used on the project were also useful, however I don't feel I have much familiarity with them because we mostly interacted with them just by running a make command. I enjoyed bringing in outside speakers to talk about their companies. I prefer it when they talk about a technical problem they faced rather than the standard "say this on a resume". I think too many companies hammer on the resume, and they all essentially say the same thing. It's much more interesting to hear about what challenges a company is facing and how they go about solving it.

----

7. I feel that you could talk about concepts vital to projects a bit earlier you talked about important ideas like virtual functions well after the project was assigned. Although I was OK with this, others who may not have understood what they needed to do may have had more trouble. The tools were generally straightforward once I knew what files to put where, although you may want to update the makefiles of later projects concerning the public test folders. I really appreciated having to learn how to code without getters and setters it helped me gain a better understanding of OOD.

----

8. I liked that this class gave me experience working with industry tools and allowed me to see how things were done in the "real world". The lectures were useful for learning c   and I feel that I have a very good understanding of it now. The papers were mostly interesting to read. The textbook reading was not useful though and I stopped doing it after the first few weeks. There sometimes felt like there was a disconnect between the projects and the lectures. The quizzes had too many trick questions. Overall great class and I think that Downing is a great lecturer.

----

9. Downing's Object Oriented Class is a great class to take if you want to learn C   and how object oriented design works in C . He is very engaging and excellent at explaining various object oriented concepts with examples. He also promotes students to get internships and get real life job experience with company talks in class. However the projects for me were pretty easy However I did have Object Oriented Experience , especially the last one. His site says 10-50 hours of work a week, but it feels like 5-12 hours. Also his tests feel unforgiving, I was presented with a bunch of blocks of code multiple choice and had to choose the right one, I was one line off and got -3 points off my final grade for a mistake that feels like -.5 to -1.

----

10. Great professor, probably the best CS professor I've had at UT so far. He's always willing to help his students and is not too strict about his syllabus. He understands the intricacies of life.

----

11. He is a great teacher. Would take again.

----

12. Good in-class structure, and good assignments challenging, but not tedious or impossible .

----

13. There's a lot to love about Downing and the course. The things I loved were lectures as they were very informative and engaging the projects, as I felt like I learned how to code well, not just "good enough" and the attentiveness of the professor as he was always willing to answer questions and or explain something again. What I didn't like were the weekly article readings, as they never helped me understand the material further. I never opened the textbook. I felt like the daily quizzes were more trick questions than tests of knowledge. Lastly, I HATED Hackerrank. I failed a project because Hackerrank is supposed to give points based on percentage of HR tests passed, but I got 2 3 tests going and only got 2 out of 40 points possible.

----

14. I had a few issues with the course. Mainly, I think there wasn't enough focus on object-oriented design. We spent a couple weeks at the beginning of the semester going over fundamentals, like pointers, exceptions, iterators, etc. These are all concepts that CS majors have covered, and I didn't think this class which is upper-division needed all the review. This caused a little bit of frustration later in the semester when there were concepts that I was hoping to focus on, but we never really did. Specifically, when planning the Darwin project, there were a lot of design decisions to be made that we hadn't thoroughly discussed, like why is using "friend" bad for encapsulation? Why grade us on our design if it was never really taught?

----

15. I enjoyed the lectures, projects, and material. It was one of the best courses I've taken at UT, and I would take it again if I could.

----

16. I enjoyed this course. All of the projects were interesting. They were not too hard or too easy. The lectures were also interesting and engaging. The only complaints that I have for this course is that there were times that information about the

projects was wrong or not explained clearly, and the book that we were supposed to read for this course did not feel incredibly relevant to what we were learning in class.

---

17. Really excellent course and best teacher I've probably ever had due to the unique teaching style and ability to explain tricky concepts. Lectures reminded me of Khan Academy videos. Tools were amazing. Projects felt a little too easy. The Collatz project was great because the simplicity of the project gave us plenty of time to learn the tools. Life project felt redundant due to its similarity to Darwin. Allocator helped me understand vectors more. Voting was very easy. Quizzes were the hardest part of the class for me and were a love-hate relationship. They were frustrating and stressful, but they got me used to testing pressure and forced me to study daily. For tests, the multiple choice format is nice because of the immediate results.

---

18. The professor is great. I do have a problem with him interrupting students mid-question all the time.

---

19. The fact that we go over the specific behaviors of C   and often relate them to other programming languages is one of the best things about this course. Having never touched C   before this course, I am coming out on the other side feeling like I have learned more about the language in use than in any other CS class I have taken. I disliked the in class quizzes, but only due to the fact that I only have my phone to do them on, which makes it hard to read questions with code in them. I do support the use of in class quizzes though, and I know I'm in the minority of not having a laptop.

---

20. I personally did not like using doxygen

---

21. I don't think our tests are fair at all. The fact that I can lose a letter grade for not remember if a return value is a reference or a value seems preposterous especially as how I can make all As on my projects even when I am working alone. I feel like I have a good grasp on the material and have done a great job on the projects, but am not very happy about my test scores.

---

22. Best CS class I have taken. I really enjoyed the practical programming advice and the Bloomberg speaker. I feel like this class has prepared me for my full time software engineering role next year.

---

23. I think the way we used gcov metrics for unit test coverage was wrong. The 'lines executed' gave the number of lines executed within the functions called through the unit tests. So, a student can technically get away with 100 line coverage by just calling the constructor in all the test cases. My suggestion is to either clarify this with students early on in the class or change the metric on which unit tests are graded upon.

---

24. I thought that the course overall was very great, but personally, I thought that one of the hardest concepts to grasp was references. When they were first introduced, I felt that the explanation of them was rather brief compared to their actual complexity. Based on other people in the class I believe that many others felt the same way especially if they, like me had little to no c   experience prior to taking this class. I would suggest possibly taking a little more time to explain references thoroughly so that in the future you won't have students still being confused over references and asking simple questions about them in class leading in to the last couple weeks of the semester. I know this isn't a class teaching c   butitwillhelp!

---

25. Great class and great professor. Only thing I didn't like was the lack of a back button on the quizzes lol.

---

26. I thoroughly enjoyed my time in this class.

---

27. 9   I loved how this class was taught. It was extremely thorough, however it felt like to me we lost a lot of time with some of the student questions. For a lot of questions when I wasn't sure how the code would work it was helpful, but sometimes people were called on to simply walk through code rather than focus on the new topic.   14   For this one I just wanted to address the project Life. Life was a great assignment but I felt like I learned the tools to do the project well after the assignment was due. I loved this class. I've selected to take software engineering next semester for the large reason that Downing will be teaching the class. Though it can feel slow, the pace we analyzed code helped me learn a lot.

---

28. Downing is a great professor. He's very knowledgeable and approachable. Sometimes he goes through material a little quickly, which can be an issue if you get hung up on something. It's not too much of a problem, as he goes over material thoroughly. The quizzes really test your knowledge, even if missing one question can mean a 40 on a quiz. Overall, I would say the experience was good.

---

29. This class introduced me to some tools that may be useful in future programming projects when collaborating with others. A downside to the requested tools but did not affect my projects was that the versions required were slightly older versions than those on my own machine. The participation. Positive - Everyone was involved, Faster than asking for someone to answer, Good proof of attendance in most cases.   My Suggestion 0 - Provide incentive on participation, Ask a question during lecture and whoever answers gets credit towards a set amount of bonus points. Ex. provide input at least 20 times, get 10 bonus points, etc.   I, myself, have heart-rate issues being in the spotlight, unless I prepare myself to face the fear. Thank You!

---

30. I liked how you made the talks extra credit. The only downside is that some people may have other classes during the time the talks are scheduled. So, they miss out on the opportunity to listen to the speakers and earn extra credit points.

---

31. Overall probably the best CS course I've taken to date at UT. My grade may be less than perfect due to my own mistakes and not anything really related to the way the course was given. I enjoyed each of the projects and felt as if they were describedvery well and graded fairly. The tools were useful, especially Google Test, Docker, and git, although I didn't fully understand the necessity of clang-format and had trouble with Doxygen at times. Atlassian was the best speaker in my opinion, although I missed Mutual Mobile.

---

32. I enjoyed going to your class every single day. There wasn't a day that I dreaded going to class and I hope to be taking your course again in the Fall!

---

33. Professor Downing was an excellent lecturer, and professor. He came to class everyday with a plan that he would make clear to us. He would encourage class participation by calling on students, and allowing students the opportunity to ask questions, which he would almost always have the answer to. The projects did a lot in teaching both OOP principles, and workflow practices that are common in the tech industry. Professor Downing helped me become a better programmer and developer.

---

34. I really enjoyed this course. The presentation style of information was very effective for me and made it easy to follow along and understand what was being taught. The quizzes could be tricky but did help highlight areas where I needed to focus on and were a good indication of some of the types of questions on the exams. I thought Travis CI and Google Test were great tools to get to use. I was not very fond of Docker and ended up not using it. I also didn't like how clang-format just destroyed the

aesthetics of my code. The projects were all worthwhile and had a clear purpose. The Allocator project was the most difficult for me. It seemed to be at pivoting point in the course and it took a bit to get a good grasp on it.

----

35. This course has been an excellent learning experience. It has given me the confidence to design better code and exposed me to some new software tools. I thought the quizzes were helpful for me to gauge how well I understood the material. One assessment that could have been improved was Test 1. Some of the questions had high weights and also follow up questions. Thus, missing one question could potentially have a significant impact on one's test grade. In contrast, I thought Test 2 balanced the number of questions and their respective weightings quite well. Lastly, I enjoyed using GitHub extensively in this class. I now understand how to use the service better and I've become more comfortable with it after taking this course.

----

36. I feel like I learned a lot of useful material in this class. The lecture style, where Downing called on random people, helped me pay attention in class more. The projects were flexible enough to where it did not take up a lot of time but it also did not feel like a blow off class. The exams were reasonable and as difficult as I would expect. My one criticism is that in the end, the extra credit blog did not really feel like it was worth the 20 points, but then again it was extra credit and I would rather have the opportunity. Overall, I enjoyed this course and I hope I can take another class with Downing in the future.
