Student comments for Arup Guha, dmarino@cs.ucf.edu

University of Central Florida
Student Perception of Instruction Comments

Year/Term: 2009/Fall
Semester: 1370

Instructor Name: Arup Guha

Department/School	Course-Section Number	Course Name
Computer Science/College of Engr & Comp Sci	COP32230003	INTRO TO PROGRAMMING WITH C

Number of Students Enrolled	Number Responding	% of Response
209	132	63.16

The thing(s) I like the MOST about this course:

Opportunities for seeking extra help was bountiful, mostly because of access to the TA's and labs outside of class hours.
Everything is explained excellently. I always understand the content of the lesson.
The teacher really defined this for course for, I've taken C++ Programming at the community college level. So I had some understanding of what to expect but the instructor really taught things in a manner that was easy to comprehend. I really enjoyed the class Mr. Guha is so far the best teacher I have had at UCF.
Honestly nothing
Programming.
Made the class fun and exciting with examples done live, not just pre-done. Assignments were fun, interesting and challenging.
The open ended problems
Candy, but seriously, Arup is just a great teacher who makes time for us.
Arup's sense of humor
Illustration of material
The T.A.'s were nice and helpful unlike T.A.'s in other classes I've had.
It concerns a topic I'm very passionate about and I've learned quite a bit.
Learning concepts through in-class example games (i.e. having the students come up and pretend to be functions). Definitely helped me understand the stuff a lot easier
Candy :3
Arup went at a steady pace and had all lectures posted online so that we could follow along.
The instructor definitely kept me interested throughout the class, and he does a good job explaining the course material considering that the course itself is fairly difficult (but not wholly difficult) if you are someone who does not have prior programming experience.
CREATIVITY OF THE INSTRUCTOR
The assignments and programs were applicable to everyday life.
Computer programming is very useful.
The professor seemed pretty enthusiastic about the material. He tried to use written and real life examples to explain the concepts of the class. He rewarded students with candy and extra credit for answering questions during review time.
The class was interesting. I wasn't too bored with it.
Although I was already experienced in programming, I feel that this course was taught very well, giving unique perspectives to enforce what I knew. The few things I did not know from this course I felt were taught very well. I have taught similar material before myself, so I know the challenges associated with teaching this material. This instructor did an exceptional job with any associated challenges with the material, and did not appear to have any problems teaching any of it. He is a great instructor!
The subject matter.
very well organized. material explained well.
How the prof. took a big class and made it simple for everyone to understand.
The programs.
I like the grading system a lot; it's nice to be challenged and have to work for higher grades. Prof. Guha is a great teacher who shows interest in the growth of his students.
Did some example problems in class
Very high quality course, you will defeinitely learn programming basics going through this class! Instructor was a good lecturer, very clear, and easier to follow than most programming professors, great notes, great programming examples. Good outside support for the course and good lab availability.
Just about every concept that was throughly explained.
The overall outline of the course was very well prepared.
Interesting
Programming. It was fun to learn how to code a program.
Arup's presentations were good. He carried himself well.
The course moved at a good pace, and the instructor was definately the highlight of this course. Top Quality Teacher.
The professor used alot of visual aids to help teach the material. he made the learning interesting by involving the class.
I'm new to programming, and I thought the pace of the course from basic to more advanced material was perfect.
I liked the structure of the course, and the assignments that were given.
I liked that Professor Guha had an excitement about him for teaching this course material.
I really enjoyed the overall class. Though it is a challenging class, the instructor made himself available for assistance making it a bit easier to understand. Also, he made the class understandable and enjoyable.
I liked the programming assignments. I wish there were more of them.
guha was great
Arup explains the concepts behind the C language very well and in depth.
I like how Arup Guha is the most personable professor here at UCF. I have taken a C++ class before this and took this class

only because I was required by my major (BSEE) but Arup Guha explained the concepts so well that all the questions I carried over from my C++ class were answered here.
The ability to interact with other students, as it was the only way I was able to learn in this class.
.
Learning to program
It was interesting.

The thing(s) I like the LEAST about this course:

Unless one had a previously vested interest in programming, this class provides very little motivation to learn. This is not by any means caused by Professor Guha; the subject material itself is the reason.
It would be nice to have a lab portion of this class to practice writing programs under the instruction of a TA.
Way to difficult for an intro level course.
The only thing that I felt wasn't up to par in this course was how much grades relied on only a few homework assignments and even fewer tests. This makes it tough to get a good solid grade if you don't really understand one topic throughly.
The pace all of the sudden picks up in the second half and if you haven't programmed before you basically can't succeed
Explanations/drawings of concepts. It just doesn't work for me. Not a huge deal though, since I knew most of the information prior.
chairs in the classroom were uncomfortable
Did not use the book, thought it was a waste of money. Professor Guha teaches well enough were i did not need the book.
The class size is just way to big.
BORING!
The way the second test was laid out was ridiculous, made it much harder than it should have been.
Most of what I didn't like was out of Arup's hands. I personally think that a computer science course doesn't belong in the Electrical Engineering curriculum and having to learn a new language like you have to in C just really doesn't appeal to someone who knows that the class will never affect his career.
Extremely fast paced, should be described as an advanced programming course.
For the students that had never used DevC++ it was rather difficult to follow and learn how to use it.
Word documents masquerading as presentations
The pace was strenuous, the tests are there to trip you up, and an ominous spike in the course's difficulty made it no where near as fun or interactive in the later parts.
Nothing
Went over examples a lot.
All the kook ass computer kids that thought they were the shit when they were dusch bags
The material was sometimes covered at a rather fast pace.
TOO HARD,,,
The teacher sometimes acted as if students were supposed to already know concepts that were just introduced.
The textbook.
I felt the material was delivered a little too quickly for my liking and that the expectations to do the assignments was too high for an "introduction" class. It almost seemed like the instruction was being catered to the people in the class that already have programming experience and as such the true beginners were left struggling to keep up. I felt the tests relied too heavily on "trick" questions especially on the multiple choice questions. I know he tried to make up for all of this with his grading scale but it was still frustrating.
Sometimes there were a few too many examples. I don't learn so much by watching someone perform a task, but by performing it myself.
The instructor's notes and lectures were so effective, I never opened my textbook the whole year. This is not bad for the course, but rather bad for me, since I spent the money for no reason.
Schedule.
assignments take way too long to do. tests are difficult.
The amount of information learned in such a short period of time.
the only complaint I have about this course are the tests. I felt that some of the later tests had problems that were very confusing and became more of guessing games rather then accurate indications of the students' abilities at programming.
Late class time
Class size was too big for this course.
Programming assignments took forever to be graded. One took almost a month! Solutions to assignments could have also been posted quicker. What kind of programming class is held in a lecture auditorium?! Find LABS UCF!!!
I was just a novice with no prior knowledge, therefore the pace of the class to be seemed a bit too rapid for my characteristics.
I feel more could have been done about the exam objectives.
I dont like programming
Strings. Strings are horrible to deal with in C. Other languages undoubtedly will handle strings better.
I think the tests can be too confusing. Like the one test were we had to pass pointers into functions. The pointers could be named a and b, while in the function they would be called b and a. This makes things unnecessarily way too confusing. I don't think a programmer would program in this fashion. Nevertheless, even if one did; the test should be more about the student understanding what's happening in the program, while keeping things un-confusing.
The material was covered very fast and the assignments seemed to jump in Difficulty and length rather than gradually get difficult.

The amount of all-or-nothing points on the programs can be rough, although it does make sense considering the amount of time given to fine tune the programs.
I disliked the pace of the course. It definitely went to fast for those who have never had any programming experience before this class.
It took a really really long time to get grades back, nearly a month for program 4. I would find it more helpful to get a grade for a previous assignment back before having to turn in the next assignment. Otherwise, I could make the same mistake on both assignments without having a chance to correct it. In short, the grading comments from one assignment can't improve the next one unless they are recieved before the next assignment is due.
tests were really hard, did not like that the multiple choice had so many "none of the above" answers
Felt tests were overly hard
For someone who isn't a CS major the math was a little confusing with little to no explanation. I guess it is assumed that everyone is CS and knows the math formulas.
There as nothing I disliked about this class.
Guha, Arup R., cannot teach he thinks that just drawing on the computer overhead is instruction, students need real instruction, i.e. explanation of functions and why they are used and when it is best to use this function of that one. Things that should be tested like basic knowledge of how functions are supposed to work were not tested. Instead he would make tests that had you figure out complicated loops and if you missed the first of the loop then the other answers would be wrong. This is an intro to C class and was taught like it was tested like it was a master¿s course. It is unfair to test a concept were if you get one question wrong you fail the test because the remaining answers are based off of the first answer. It is incomprehensible that an instructor should have to curve so much so that 80 plus percent of the students do not fail. I.e. a 50 is a C? This is not a failure of the students but of the instructors direct ability to determine what is necessary and how to teach it so that students that are just coming into the field can relate and logically understand what they to in order to pass this course with a real grade.
.
Informal
Spent the same amount of time on things that were simple and easy to learn as the incredibly hard parts.

What is your reaction to the method of evaluating your mastery of the course (i.e., testing, grading, out of class assignments (term papers), instructor feedback, etc.):

Tests and assignments were graded in harmony to how large and intimidating the work actually is, and thus seems quite satisfactory.
I think part of our exams should be done in a computer lab.
As I stated above, I feel as if the grade is weighed to heavily I only a few separate grades. I would have preferred it if there were more homework assignments. Overall though I felt it was a pretty even grading scale.
this class could use more grades (tests) and the number of programs was good.
Programming and tests seemed to be a decent balance.
Fair. I think the assignments should be worth more.
N/a
You should make the "A" curve from 85-100, 85-87 being A-. I took like all the tests from your previous years and they are so much easier. I feel like we got the short end of the stick. I just hope to god I end up with an "A". Seriously though right now the class average is a 63.
not a huge fan of the last question of the test being a ridiculously hard one that maybe 5% of the class could get.
Fair
Very content with his grading methods. Very fair and understanding.
tests requiring handwritten response for programs should not be used in a computer class Never received any feedback from instructor
The tests were ridiculously difficult. If not for the curves, plenty of people who truly work to their ability would be borderline failing.
The programs we were given helped me understand the way to program a lot more than what a book would say. Done pretty well
Programming assignments and tests fairly evauluated our mastery of the course. Basically, if you did the programming assignments, and spent several hours trying to figure them out then you learned what you needed to know for test.
TESTING, AND ASSIGNMENTS
Programs were fair. The tests were rather hard, and I did not agree with the fact that about 8/20 of the correct answers to the multiple choice section of the test were "none of the above."
Pretty good.
The tests were fair, I think. I feel confident with the skills I've acquired in the course, and my grades reflect that.
The tests and homework assignments were very effective in testing our ability in the course. The grading and feedback were just as effective in showing how well we did on each assignment/test.
The exams were challenging but fair. I hadn't taken programming before, but felt that everything on the exams we were prepared for.
tests are difficult, assignments take a long time, grading is very forgiving
I was very impressed with the testing.
The tests were very difficult, but would be fine if they were not weighted as high.
Decent feedback was given for programming assignments. I appreciated that I was told exactly why I would lose points if any were taken off.
The assignments due dates were too soon. He needs to slow the lectures down just a tad bit
Test questions became unnecessarily tricky at times. Never cared for the free response sections of tests. Some programming assignments seemed extremely tough for an "intro" programming class.
Very effective.
fair
Interesting. The combination of actual coding, multiple choice tests, and having to hand write code seemed like a much broader way of getting a feel for our understanding of the work than most courses.
Can't really complain all that much. It would be nicer, if the programs were graded more.
The programs were excellent material to reflect what we were learning in class. If you couldnt write the program you knew you needed to study more.
The test were very difficult.
I think the combination of programs and exams works great.
A majority of the multiple choice questions on the tests required the student to keep track of many numbers in loops and perform iterative functions excessively; there is a point where a question strays from testing a student's knowledge and understanding concepts(how loops work, and what they are used for), and focuses more on the student's ability to perform many calculations.
I think that his assignments were really hard. This is good thing and a bad thing. It is a good thing because it made me really try hard to grasp the concepts of the material that was required for the assignment. It is also a bad thing because some of the stuff was just way beyond my comprehension and would not have been possible to complete without getting any assistance / help whatsoever.

The grading on assignments was fair though I would have appreciated the programs that we were suppose to write, to have been graded a bit faster than they were.
I know it's hard to test programming skills, but handwriting code really doesn't simulate a real programming situation. It would be really cool if a computerized programming test could be developed where students would have to write and submit a program in a certain amount of time and couldn't navigate away from the page for outside help. Could that be done through webcourses?
I though the test were a fair method of testing our mastery of the subject.
Evaluation is not very practical the projects could be accomplished; however, they were far past learning basic concepts and would take vastly longer times than what he would expect. The tests were not anything like what his examples were like a no real way to study for them they felt like they were exercises in trying not to get tricked, who uses does not exist for multiple choice answers especially when you are passing answers for test questions and one answer relies on the next.
.
Fair
It was all good.

Additional comments and suggestions for improvement:

To garner more motivation into learning programming subjects, perhaps a less derivative teaching method should be suggested to the professors? I realize that the professors are those with the knowledge of how programming should be taught and what to teach, but maybe researching an alternative, more engaging method may yeild higher interest in the subject.
This class as been by far my favorite at UCF so far. Mr. Guha really made this class worthwhile.
i suggest that there should definitely be more tests and not just 2 and a final, to me that doesn't reflect the overall effort of a student
You're a fantastic instructor, I hope to have you as my professor later in my C.S. career.
N/a
Great teacher with hard tests.
Suggest to require a prerequisite programming course (2000 level) be taken prior to this course. Additionally the title"Introduction to C programming" is deceiving. A more appropriate and accurate title would be "C Programming for Scientists and Engineers".
Tests should be either multiple choice, or taken in a computer lab, or dropped altogether and replaced by longer assignments.
Demonstrations with teh students acting teh parts of computer memory are completely ineffective
Keep up the good work!
Moar candy
FUCK C PROGRAMING
Thanks for being such a great teacher, and giving me so much feed back on my papers!
EASIER PROGRAMS
If the teacher is going to have hard tests and tricky questions, I don't agree with him consistently making the correct answer "none of the above." This only adds confusion to the already tricky test, and even if the student found the correct answer, they may me skeptical to answer "none of the above" for almost half of the test.
None.
Cool beans :)
Thank you Dr. Guha! You really made this class fun and worthwhile!
n/a
Make programs worth more.
It is important to explain why to use certain codes in certain situations.
Not sure why C is the language of choice, but I would have liked to start off in Java, C++, or even C#. Find a LAB!!!
None.
I thoroughly enjoyed your course. I look forward to having you again as my professor.
N/A
Make the tests a bit less confusing. This doesn't mean easier. Just less confusing. How about the option of dropping one of the program assignment grades and doing an optional program to replace it. So if you've messed up on one, you can justify yourself.
My suggestion is an idea, which came from noticing half the class already knew everything and the other half was just learning everything for the first time. My idea is as follows: possibly having some kind of test that will allow those who pass the test to skip this intro class, which will then allow the intro class to focus more on the needs on those just starting to program for the first time, instead of forcing to move on to give equal attention to everything.
Overall, the class lectures were very clear and helpful. However, sometimes staring at the projected code on the screen gets really really dull. I found it much easier to pay attention when you were drawing visual representations of the code and even writing the code out by hand.
I have no other additional comments.
On one of the projects for a dictionary some common words were not used for his example but nigger and fag were, the man is not very politically correct.
.

University of Central Florida — Year/Term: 2009/Fall
Student Perception of Instruction Comments — Semester: 1370

Instructor Name: Arup Guha

Department/School	Course-Section Number	Course Name
Computer Science/College of Engr & Comp Sci	CIS33620001	CRYPTOGRAPHY AND INFO SECURITY

Number of Students Enrolled	Number Responding	% of Response
36	24	66.67

The thing(s) I like the MOST about this course:

What we learned in class and diffie hellman. That was the best method that i have learned.
Arup's general enthusiasm for learning and knowledge spills over into the course and is infectious.
The math of it was very good.
Professor Guha is always a pleasure to have. He loves the material and it shows in the care he takes creating homework projects and in grading. Overall I learned a lot in this class and although it was tough I wouldn't have it any other way.
I loved having our tests split up into quizzes instead. It really helped eased the pressure in the class and made it much more manageable to study for. I also really liked cracking the hand ciphers in the beginning of the course.
Interesting material that was very relevant to the Computer Science major.
no final
Arup was the best instructor I have had my entire college experience. Too bad it took until my senior year for it to happen.
The Content
Some of the concepts were quite interesting. The instructor seemed generally enthusiastic about teaching the material.
The instructor's explanation and method of teaching each chapter was perfect. This made the knowledge more understandable and enjoying to learn.
Excellent teacher really knows his stuff, Too fast paced for me however

The thing(s) I like the LEAST about this course:

Some methods he went through it to fast so most of it we had to learn on our own and some of the homework was really difficult and long.
bad grading criteria, tegrity lectures - biggest scam in the world
The beginning with doing decryption by hand. Found it to be really tedious, maybe ease up on that a bit.
The availability of online material was lacking in the final month or so of class.
Towards the end of the course, there were more programming assignments, which wasn't as big of an issue for me as it was for some of my classmates. I think Arup may have forgotten that not everyone in the class had a strong programming background- the assignments were given with little explanation, so it was difficult to do if you did not already have a good handle on programming. For those students with less expereince, it would have been helpful to have some hints or a rough guideline to help them start off their projects.
While I would argue that Professor Guha is probably the most student-concerned professor I've ever had this semester I thought there was a little bit of lag between course expectations and presentation of material. The online components provided an excellent alternative at first however when students began to become reliant on it the system failed. Though Professor Guha did attempt to get it back online again, homework and quizzes were assigned that was on material not covered for the online students.
tough quizzes
I would have liked to have a little more programming involved in the course. I understand why there wasn't though.
The online section never worked
I think the workload for this course was a bit much for a 3000-level course. I did double to triple the amount of work for this course that I did for some of my 4000-level courses. At times I felt as though I were taking a number theory course. I realize that some of the encryption standards require some advanced math concepts, but I felt that having to do proofs of those concepts was a bit much. In other security courses I've taken, I don't recall having to rely so heavily on the background of the math itself.
There should be options to redo assignments when out sick. However, the partial credit I received was very appreciated.
The use of Tegrity. I have nothing against online courses with video, but Tegrity performs very poorly

What is your reaction to the method of evaluating your mastery of the course (i.e., testing, grading, out of class assignments (term papers), instructor feedback, etc.):

This was all fairly done.
I found the quiz system to be much more effective in me retaining the knowledge and never let myself become too lax with studying. Kept the pace up all the time and I think that if you do poorly on a quiz there is still ample chance to recover.
Well done. I really like the format of 8 quiz/tests and 8 homework assignments with no final.
As I said before, I liked having our tests split up into quizzes. The homework was occassionally long, but more typically was a very reasonable length and we were given plenty of time to finish it. Arup is also very good about getting grades back to students quickly, so we were never kept in the dark about how we did on a quiz or homework assignment.
The quizzes and homework were very relevant to the material though, as stated above, the material was not always available for those taking the online component.
The quizzes were hard, but never seemed unfair.
I didn't particular like having so many different quizzes. The instructions on the quiz were quite vague. We would often discover the instructor's true intent for a question only after the quizzes were graded. It also didn't help that the instructor was not present during the majority of the time of the quizzes. The quizzes sometimes required us to do calculations that were unnecessarily overcomplicated. For example, calculating the mods of large exponential numbers. Grades based on our ability to do these calculations manually did not really evaluate our understanding of the cryptography concept itself. The homework assignments tended to be quite demanding as well.
The testing was perfect, not too easy while not too hard. However, there were some parts on individual quizes that weren't discussed before the quiz was given.

Additional comments and suggestions for improvement:

He is a great professor overall
Arup is one of the best teachers at UCF. Tests can be hard sometimes but if you know the information you'll do great.
N/A
The only thing that I think could be improved is that Arup sometimes goes on really long tangents on unrelated subjects, so every now and then an entire class period was blown on somthing way off topic.
The online component has promise but only if it correctly utilized. If it can't be then drop it entirely.
Maybe slightly less math, a little more conceptual material and testing.
Arup is by far the best professor in the CS department!! Please teach more classes. Your method of teaching is great.

University of Central Florida
Student Perception of Instruction Comments

Year/Term: 2009/Fall
Semester: 1370

Instructor Name: Arup Guha

Department/School	Course-Section Number	Course Name
Computer Science/College of Engr & Comp Sci	COT39600001	CS FOUNDATION EXAM

Number of Students Enrolled	Number Responding	% of Response
46	29	63.04

The thing(s) I like the MOST about this course:

I passed this time!
Took this Spring '09 and passed.
N/A This was the foundation exam
That it weeds people out, and that I passed it on the first try

The thing(s) I like the LEAST about this course:

I had to take it twice
The room used for this test is HORRIBLE! The chairs are slated forward and are extremely uncomfortable, the worst room in the whole University!
Took this Spring '09 and passed.
N/A This was the foundation exam
The whole class rides on this one test!
That as of the time after i did it, they allowed people to have a cheat sheet

What is your reaction to the method of evaluating your mastery of the course (i.e., testing, grading, out of class assignments (term papers), instructor feedback, etc.):

The test can be a little long for one setting, my head was throbbing when I was finished.
Took this Spring '09 and passed.
N/A This was the foundation exam
It works I suppose.
good. just no more cheat sheets

Additional comments and suggestions for improvement:

Arup Guha is the greatest teacher ever!
Took this Spring '09 and passed.
Not sure why I am filling out this for the Foundation Exam, but here it is.
For a token class, I'm not really sure how to evaluate it.
keep it hard.

University of Central Florida — Year/Term: 2009/Fall
Student Perception of Instruction Comments — Semester: 1370

Instructor Name: Arup Guha

Computer Science/College of Engr & Comp Sci	COP3503H0201	HONORS COMPUTER SCIENCE II
Department/School	**Course-Section Number**	**Course Name**

21	13	61.90
Number of Students Enrolled	**Number Responding**	**% of Response**

The thing(s) I like the MOST about this course:

The course material
That we have to do a project for the whole semester.
Interesting Ideas
Arup is one of UCF's most valuable faculty member.
Arup Guha is an amazing professor. His lectures are always entertaining and engaging and his interaction with the class excellent. Should you miss a class or want to study for a test, he maintains a website with notes from current and past years as well as past tests and code samples and etc. It proves to be a very effective complement to his lectures. I also appreciate the open nature of the final project which, in the first week, allows students to form into groups and to then work on a project that is related to computer science and that requires programming- as long as the project conforms to those two criteria and receives Professor Guha's approval then it's acceptable and the group just needs to show that they put significant effort into it. It's a nice way to allow computer science majors to work on something that interests them rather than to just continue to produce a bunch of mini programs.
Little homework, good pace
I liked the course material, and the way that it was presented in class. I liked that we were able to chose a program to work on. We had a lot of fun with the project, and I was able to learn a new programing language.
The pace of work and course work was easily adapted to our class, and the professor had a genuine interest in our learning ability. he made the course work fun and easy to learn.
Very relaxed, group project over annoying assignments
Project presentations
Arup!

The thing(s) I like the LEAST about this course:

Everything was pretty good.
Although the ideas were interesting, they were never really taught to us very well. Also, the lack of professionalism by showing up late to each and every class, including test days was exceptionally frustrating.
n/a
The tests can be brutal because the class I took was only 50 minutes per a class. When I took CS1 I had an hour and a half per a class. However, I did not notice much difference between test lengths, and either way there are some questions that can take quite a bit longer for a particular student no matter how you slice it. On average in CS1, I completed my tests well before time was up, but I spent a vast majority of that time on one question and relatively little on the others. That didn't change for me in CS2 and while I plowed through the first test faster than anyone else and received an excellent grade, on the second test I didn't even finish it. My second gripe is with final project groups which I will discuss below in "suggestions for improvement".
Summation question on the first test
I wish we could have done more code examples and programs. I had trouble keeping up with the pace of the course.
nothing
AM class

What is your reaction to the method of evaluating your mastery of the course (i.e., testing, grading, out of class assignments (term papers), instructor feedback, etc.):

The tests were fair. Everything was very fair.
Not very fair considering that the material wasn't really gone over well.
Great, even better than the normal section.
Evaluation is based primarily on three tests and the final project. The final project is still a work in progress so I can't give feedback on the evaluation for that, but I am not a fan of test-based evaluation. Professor Guha does a good job of grading the tests, but I tend to find that in computer science, for whatever reason, it is often an occurrence that despite having a small amount of notes accessible, I will completely brain fart on a single specific of a concept that I otherwise know extremely well. In the second test, this happened with a concept that I knew supremely well, but for whatever reason brain farted wholly on one specific of it and lost loads of points as a result. Problem is, this grade did not reflect my actual understanding of that topic.
Good enough
The tests and assignments were very fair.
We were graded promptly and in a manner which suited the material.
Very good
Tests were prepared well

Additional comments and suggestions for improvement:

Treat your students like they are not on the programming team, as most of them are not.
Raises.
My biggest suggestion for improvement is to implement some professor-supervised system of selecting formal group leaders for final project groups. I had the unfortunate situation of coming into my group as "the fourth man". However, the project was a small video game and by hobby and increasingly by trade as well I am a video game designer with years of experience developing even larger-scale projects. So, without meaning to sound arrogant, I could easily have gotten the project on the right track. The problem was that, as the "fourth man", I didn't want to barge in and take control for obvious social reasons. So while I contributed plenty of expertise in many areas, nearly none of it actually mattered because the two founders of the group had decided to do a majority of the main code base yet they didn't really know what they were doing nor did they really put in enough time and/or do enough research to rectify that. They were also a bit gruff about it and so any advice I could have offered to help them was nearly immediately turned-down. Worse, on my end and that of the third group member's end, was that we continually got conflicting and/or impossible requests from the other two group members. They also delayed working on the project as much as possible and because of such held everything up until the final minute, whence everything became an all-out scramble to piece together something of relative quality. I feel that formal group leaders would help rectify this situation as it would force groups to consider who would best lead the group. In this situation, I may have been put in charge and would not have had the issue I had with leveraging my experience. Or, one of the founding members may have been the group leader, and then when everything that has happened did, that founding group member would be unable to shift blame and responsibility (while not actually shifting responsibility). More-likely, that group member would not even try to do such because the results of any such attempts would just land at that person's feet. As things stand currently, the third group member and myself are presently contemplating just splintering from the other two group members and developing our own code base because we are not confident that the other two members' code base will actually work or get done because currently that is not the case on either front and if we go forth with splinter code they aren't going to integrate it into their code because they will have no code to integrate into and will have effectively done zero work for the project.

University of Central Florida — Year/Term: 2009/Fall
Student Perception of Instruction Comments — Semester: 1370

Instructor Name: Arup Guha

Computer Science/College of Engr & Comp Sci	COP3930H0201	HON SPECIAL TOPIC
Department/School	**Course-Section Number**	**Course Name**

11	8	72.73
Number of Students Enrolled	**Number Responding**	**% of Response**

The thing(s) I like the MOST about this course:

Purely on Arup's part it was an amaligimation of the class participation lessons from various disciplines. This allowed a hands-on approach to learning about a subject, different and imho awesome. Dr. KV was very fast in his math and often lost me by assuming we knew differential equations when many of us didn't.
It wasn't the same thing every class and you generally didn't know what to expect. We got to do things like play rock, paper, scissors, and get married for the sake of learning.
I liked how we learned both the topics and the applications of the theory. I also like how the course was not just a continuous building upon each each topic. Rather it was more like an interesting special topics course. It kept the material from becoming boring.
A lot of the topics are very interesting, and both professors are quite good at teaching.

The thing(s) I like the LEAST about this course:

STRUCTURE! Don't ever again assign a massive project in the last 5 weeks of a course. I'm pretty sure its unconstitutional "cruel and unusual" ;)
Differential equations without having taken differential equations =(
The paper.
The final project is killing me. It seems to have too high an expectation for what the course is.
Occasionally the assignments were a little bit unclear. Mostly, though, the issue I had was that a project was introduced at the beginning of the semester as being a major project that would probably take up most of the semester. It was then assigned about a month or so before the end of the semester, with what seemed to be the same high expectations. Also, we are expected to understand calculus and some differential equations, though neither of those classes were pre-reqs for the course, and it seems only a few people have actually taken dif eq yet. When honors students sign up for honors seminars, we are encouraged to sign up for things outside of our usual disciplines, but that would be almost impossible with a course like this. I know it is the first time it has been taught, so these are mostly just suggestions for fixing it for next time.

What is your reaction to the method of evaluating your mastery of the course (i.e., testing, grading, out of class assignments (term papers), instructor feedback, etc.):

Testing was eclectic due to the vastly different teaching styles and apparent disparity in what the class was supposed to be covering.
It's not too bad, although the test was strange because KV grades it entirely based on doing all the right steps to problems we've done before and Arup gives problems that make us think.
The homeworks were all good at testing the concepts taught in class. I felt the exams were hard due to the amount covered in the class.
It seems to be fair and acceptable in relation to the class work.

Additional comments and suggestions for improvement:

I would either recommend a slightly smaller capstone project, or have students start on it earlier. I would also try to have some pre-recs to the course. People who weren't Comp Sci, Engineering or Math/Physics majors seemed to have difficulty keeping up with course work.