From: **Student Rating of Instruction System** sri@uvu.edu
Subject: Course Evaluation Results for: CS 339R 001 - Charles D. Allison
Date: May 11, 2012 at 10:33 AM
To: <10005194@uvu.edu> 10005194@uvu.edu

Course Evaluation Report

Dear Faculty Member,

The Student Rating of Instruction system is now closed for the courses listed and your grades should all be submitted. If they are not, please work with the registrar's office immediately to submit your grades. Your detailed survey results are shown below.

Term	Division	Department	Course ID	Course	Description	Professor	Evaluations Taken	Total Enrollment	% Complete
201220	TC	CSE	CS 339R 001	21002	Adv Programming Lang Python	Charles D. Allison	29	33	87.9

Demographics

Total 29

Description	Response Total	Response Percent
Class Standing		
Senior	22	76%
Junior	6	21%
Graduate	1	3%
Crse req'd for program?		
Elective	17	59%
Both	7	24%
Required	4	14%
Crse Requirement		
My Major	25	86%
Both	1	3%

Instructor

Total	Strongly Agree	Agree(%)	Neutral(%)	Disagree(%)	Strongly Disagree(%)	Avg	Std Dev
29	87	11	1	0	1	4.83	0.30

Description	Total	Strongly Agree	Agree(%)	Neutral(%)	Disagree(%)	Strongly Disagree(%)	Avg	Std Dev
Organized	**29**	86	7	7			**4.79**	0.56
Respectful	**29**	86	14				**4.86**	0.35
Fair	**29**	83	14			3	**4.72**	0.80
Clarity	**29**	83	14	3			**4.79**	0.49
Knowledgeable	**29**	97	3				**4.97**	0.19
Timely Feedback	**29**	86	14				**4.86**	0.35
Achievement Standards	**29**	90	10				**4.90**	0.31
Recommend	**29**	83	14			3	**4.72**	0.80

Comments

Total 29

Description
Helpful
Chuck is open to talking about any subject that is related to the class, this helps students to open up and criticize/critique his in class code. Chuck will usually shoot down the person criticizing his code when he knows that the suggestion is ill advised or just plane wrong. he does this in a playful/friendly kind of way though. when the commenter happens to be right chuck is not afraid to admit that their view is better, or even in unknown areas will admit that he doesn't know enough.. and then he researches the question and updates the class the next time we meet.
Clear expectations on what needs to be done and when. All information is available online. Attendance is not required! :)
He is an expert on the course material and explains the hows and whys, not just the whats. I thought it was great how he would explain how the various functions of the language worked. We went into more depth than usual classes. I also liked how we went over so much material. From the basics to integrated db handling and web crawling, it was great. It definitely wasn't an easy class, but it was always fair and

challenging enough to make you grow. This is the kind of class where you really get your money's worth. I wish I could take more classes from this professor. You should give him a raise.

He would take time to recognize student concerns, explain in detail concepts, and had a general desire to help students learn.

I found helpful that he is knowledgeable about the topic, which gives him the freedom to tricky the student.

I found this course to be one of the most challenging but fruitful programming classes I have taken.

I have very little to say other than that he's a very good teacher. There were enough assignments and quizzes to keep me on my toes so I never forgot about the class and set it aside.

I really liked the programs, in particular the web crawler, and the file zipper. The lectures were really good too.

Like I said about 3370, the quizzes and code examples were very very helpful in sinking the material in. The projects covered the intended material well, and I definitely feel I got a lot out of the class even if I knew quite a bit of Python before it. The assignments were nicely paced too - I didn't have trouble keeping up along with my other 3 classes at the same time. (I don't know if all other students would say the same though, since Python is so easy for me).

Prof Allison's obvious enthusiasm for the course is contagious. His pre-class preparation was apparent.

Prof. Allison as always gives clear examples and is responsive to questions. I always enjoy his classes.

Professor Allison is an amazing teacher. I recommend him to all the CS students at UVU. The course is well-structured, clear, and challenging. Highly recommended!

Professor Allison is one of the most knowledgeable and passionate teacher at UVU. I honestly have a hard time in programming but professor Allison makes it interesting and his classes are always very motivating.

Professor Allison is passionate about Python. Because of this he is able to make class exciting. One thing that I appreciated was the constant reminders that he gave on Canvas. I also noticed that throughout the semester he was constantly adding new code and improving the curriculum. Because of the high standard in the class I feel like I learned more. The reason I chose to take this course (as opposed to another elective) was that I new it would be challenging and that I would learn a lot.

The extra credit quizzes helped refresh previous topics

This was one of my favorite classes I have taken. Professor Allison is excellent at teaching Python. As someone who had not written a single line of Python code before the semester, I found the assignments challenging but not overwhelming (some of them were even fun to figure out). Python has quickly become one of my favorite languages and I now use it regularly in my job (GIS systems). I have highly recommend this class to several of my friends.

Very good course. I learned a lot.

Very helpful during office hours, treated students kindly. Knows his stuff.

We were instructed very well. I did like the difficulty of the tests, I would much rather work for my grade.

it broke away from the standard c++ classes and gave me something different to sink my teeth into. still not as easy to code as VB, but getting there, definitely.

use of canvas and easy access to slides and example code.

Suggestions

Give this man a raise, great professor!

I would have *really* liked learning about GUI development, but everything else was very useful too, so I don't know what you could swap out. I also would suggest posting the quizzes and their answers on canvas so we can go to those to review along with everything else.

I would like several smaller assignments in addition to the large assignments. That way i can learn smaller concepts before implementing them in the large assignments. If that makes any sense...

In this class specifically, not generally, the work load was a little light. Could have done more assignments and covered more material (i.e. GUI's)

It'd be really cool if the files were sorted by date. I think assignments should be worth a higher percentage. Assignments should be more frequent. Have quizzes be online, harder, but open book.

Keep it simple about learning a language and it's constructs. No GUIs please.

Maybe not so tough on the test or allow for two sided piece of paper at least (after all, in the real world we would have access to web and books). I think I would have still

learned as much and maybe not dreaded the exams as much. (4hrs+ for each of the two midterms...not complaining just saying)
My only suggestion would be for you to pay a little more attention when creating/uploading homework assignments; seems more often than not I have to go back and rework an assignment because something was left out that you later decided was important, or that you uploaded the wrong files, etc.
One suggestion I have is that we could have some lectures assigned to home reading and then we could develop some small projects together in class in order to get more hands on with the instructor to build knowledge.
Tests are way too drawn out and difficult. Sure, we all pass (pretty much), but after a few hours, it gets difficult to focus while testing. I would prefer more shorter tests to infrequent leviathans.
The assignments could have been more challenging. Remember, Python makes everything much easier!
This instructor should teach the topics, before tricking on the topics. He teaches how to solve tricky problems and not the main topics of class. Should stay focus with the class main objectives, which is definitely not how to solve tricky problems.
This is nitpicking because I thought the class was awesome. A couple things I would have liked to do as well though are: an assignment with a gui, one with a packaged python program into an exe, and one that used threads and multiprocessing. Having said that though, it would have been really hard to squeeze those things in without making the class too big.
This is one of the fairest classes that I've taken all year. I don't really have any suggestions at all.
Towards the beginning of the semester I was a little annoyed with emphasis/talk on how many lines of code an assignment should be. Later on I began to see that lines of code is a good indication of how pythonic my code is. That I need to watch for code that can be converted to pythonic methods, such as loops that can be represented as comprehension. Would have liked to have been able to use book on tests :)
Would have loved to cover GUIs and Concurrency
go over the language/class structure more (special class things/structures are covered but not always needed in the projects, yet it seems to be a bigger part of testing) for example the __X__ thing that specifies what can inherit from the given class
instead of two very long tests (I never spent less than 2 1/2 hours on the ones in this class), I would prefer four tests of half the length. I feel the length of the tests was excessive, and it was really difficult to find time outside class to take them because of the time expectations
keep up the good work, maybe some more coding, I didn't feel like we had to work very hard.
keep up the good work.
more or slightly more extensive assignments so that we can have practice with more of the material.

You can also view your results in UVLink on your faculty detail schedule. Please reply to this e-mail with any questions or concerns.

Thank you!

From: **Student Rating of Instruction System** sri@uvu.edu
Subject: Course Evaluation Results for: CS 339R 001 - Charles D. Allison
Date: May 11, 2011 at 6:34 PM
To: 10005194@uvu.edu

Course Evaluation Report

Dear Faculty Member,

The Student Rating of Instruction system is now closed, and your grades should all be submitted. If they are not, please work with the registrar's office immediately TO submit your grades. Your detailed survey results are shown below.

Term	Division	Department	Course ID	Course	Description	Professor	Evaluations Taken	Total Enrollment	% Complete
201120	TC	CSE	CS 339R 001	21002	Adv Programming Language Other	Charles D. Allison	26	32	81.3

Demographics

Total 26

Description	Response Total	Response Percent
Class Standing		
Senior	17	65%
Junior	6	23%
Freshman	3	12%
Crse req'd for program?		
Elective	20	77%
Both	5	19%
Required	1	4%
Crse Requirement		
My Major	22	85%
General Education	2	8%

Instructor

Total	Strongly Agree	Agree(%)	Neutral(%)	Disagree(%)	Strongly Disagree(%)	Avg	Std Dev
26	91	8	0	0	0	4.91	0.22

Description	Total	Strongly Agree	Agree(%)	Neutral(%)	Disagree(%)	Strongly Disagree(%)	Avg	Std Dev
Organized	26	81	19				4.81	0.40
Respectful	26	96	4				4.96	0.20
Fair	26	88	12				4.88	0.33
Clarity	26	92	4	4			4.88	0.43
Knowledgeable	26	100					5.00	0.00
Timely Feedback	26	92	8				4.92	0.27
Achievement Standards	26	92	8				4.92	0.27
Recommend	26	88	12				4.88	0.33

Comments

Total 26

Description
Helpful
At least the IEEE FP info at the beginning of this class should be required for all students. The rest was interesting and good, but I may not use it as much in my career. Still it was good to take this class.
Chuck is a great teacher, and I've always enjoyed taking classes from him. I feel like I have a good understanding of python now, and would be able to quickly get up to speed in a job that required the knowledge of this language.
Difficult class, but was lots of fun. Really explored some of the finer points of Python. I learned a lot about the language and programming in general -- would recommend to anyone!
Everything. Especially his prolific assistance and updates through email.
He is very knowledgeable on the subject and can answer most questions that the students have, and if he doesn't know the answer at that time he latter finds it and presents it to the class.
He is very personable in front of the class. Is willing to show more examples or find and answer to question immediately. He is also very passionate about teaching and is able to

answer to question immediately. He is also very passionate about teaching and is able to make every subject he teaches more interesting.

He was very willing to help me outside of the class. I would have never made it through the class with out his help. He made the class interesting and related well with all of the students. He is very patient answering questions which students ask.

He's fair and courteous, and knows a lot about the subject.

I liked how enthusiastic the instructor was about the subject. Made keeping interest in class easier.

I liked the way the class was prepared and how all material was available from the website.

Instructor very knowledgable. Able to answer questions and explain problems in most cases. Interactive with class through instruction and email.

It was an unusually comprehensive overview of the language; most language classes I have taken spend a lot of time on a few things, this class spent a little time on a lot of things, which meant I learned a lot of new things about the language.

It's a good introduction to python. The teacher gave plenty of examples on each of the topics and was willing to answer any questions students might've had.

Now I know Python! YAY! Great learning experience and I feel I picked up on the language quite quickly due to your teaching style and the assignments given.

Prof Allison's abilities to show example code, provide enthusiasm for great features of the language, give timely feedback about student performance, and expect high standards are amazing! He really knows how to TEACH. He doesn't lecture, preach, whine, or become monotonous. He has a great ability to listen to all student's questions, no matter the level of the student. He treats the most intelligent to the ones with budding intelligence the same. I recommend this teacher to my friends.

Professor Allison has an amazing ability to be concise and accurate, which is probably why there is always room for a bit of humor in the class. His attempts at engaging the students are more often successful than not.

Professor Allison is very experienced in the field of computer science. I find it extremely helpful his ability to relate concepts and paradigms from other computer languages to Python. Ordinarily, I would say that he did a very good job teaching this course. However, since it was his first time teaching this particular subject, I would have to say he did an outstanding job. There were a few things he mentioned in class that really stood out. It helped me to a greater understanding of the Python language and computer science in general. Some specific things that come to mind: (most)Design Patterns exist to work around the restrictions of a strongly typed language. The type of 'type' is type. This helped me to better understand the notion of everything in Python is an object. There are no meta-meta-classes. I had wondered from time to time if there might be (but I could not think of any application for such). Some day I would like to sit with him and chat about math and such

The class is good. I decided I don't like Python very much because I am not good at it, but I still learned a ton in the class and it was well worth my time (and money). It was the first time the class was offered, but it didn't feel like it. Dr. Allison was always well prepared.

This course was better presented than the other Elective Languages classes. The depth at which the course material covered greatly prepared me to truly understand how python works and how best to use it in the real world.

Very willing to help address students during office hours.

good teacher student relationship. he doesn't make you feel like you are putting him out when you ask for help.

Suggestions

Always keep in touch with what gives you the ability treat all the students the same. Never lose your ability share your enthusiasm for programming and knowledge. Keep up the great teaching!

First time through, and it was excellent. We didn't cover any of the things I wanted to learn, but I'm not disappointed. Would have liked to explore web/network python programming, as well as GUI. Maybe even some multi-core/machine type projects.

Hire a grader that can put detailed feedback in the assignments when they are turned back or graded. I know you don't have time to give detailed feedback on everything, but a grader should. That would have been the most helpful thing.

I felt we had way to many homeworks that covered list comprehension. It would be nice to have the homeworks cover a wider variety of the topics we are actually studying instead of so many list comprehension problems. Also the first test had like three or four questions regarding specific built in functions and it would have great to know that we were going to be tested heavily on our knowledge of those specific functions.

I really liked program 3 and how it could really relate to day to day needs in the work place. If all the programs could be similar (program 2), I think it would add more value to the course.

I would like to see GUI covered in this class.

Keep it up.

Keep on keepin' on.
Make Python seem harder, so students will feel like it's actually a programming language. :)
Make the tests easier haha
More Assignments, please!
Naught.
None really.
None.
Personally, I prefer the O'Reilly books. This (Beazley) book is a good compliment to the O'Reilly books, but I find it easier to learn from O'Reilly, especially "Learning Python" by Lutz. Oftentimes Beazley will give code examples, but not show the output. Lutz gives numerous, appropriate code examples with the output. Still, the Beazley book was not bad. And it came recommended by Guido. Beazley is especially good about exposing many of the modules and their methods in Python's standard library. This is noticeably lacking in Lutz. Other than that, my only request would be to have the slides available a little sooner. I like to print copies of them the night before class so I can have them in front of me during the lecture. Sometimes the slides were prepared late the previous evening and I could not print them before class. This goes along with the first time a course is taught I suppose. It is harder to take notes if I can not annotate a slide that is being discussed.
This class is intense. Maybe you could drop a topic or so and take it slower.
This is just me, but it went a little too fast. Perhaps that was due to my schedule though. Everyone else seemed to be doing just fine. Watch out for strugglers, they need all the help they can get.
Try to be more clear what is going to be expected on the tests, you don't have to give away the subjects but to know what to expect is nice.

You can also VIEW your results IN UVLink. Please reply TO this E-mail WITH ANY questions OR concerns.

Thank you!