THE UNIVERSITY OF WESTERN AUSTRALIA

Students' Unit Reflective Feedback

Unit title: **Systems Programming** Unit code: **CITS2002**
Campus: **Crawley** Teaching period: **SEM-2 2016**
Start Date: **01/08/2016** End Date: **26/11/2016**

Faculty: **Engineering, Computing and Mathematics**
School: **Computer Science and Software Engineering**

Comments posted by students

[1] for the assessment is not clearly state whats the purpose and the real question of the assessment. for me is hard to understand the question it took me like 10 time reading with googling. the content is fine.

[2] Assignment outline was still being edited after it was brought out, this created a little bit of confusion but wasn't detrimental to anything already created.

[3] The outline for project 2 was overly vague and ambiguously worded, my partner and I had no idea of what was expected of us

[4] This unit has been a rich and engaging learning experience. I have learned a lot and have been challenged to apply what I have learned in interesting new ways. Chris is an excellent lecturer and his deep understanding of the subject matter is obvious and he is willing and able to answer any and all questions during or after the lectures. The assignments were challenging, interesting and engaging and really cemented my understanding of the subject matter and showed me how what I learned could be applied. Thankyou Chris!

[5] The main source of material for this unit were the lecture slides, and I found that they often did not match up with what was actually covered in lectures (mostly the lecture slides were much more broader and covered more topics than what were covered in lectures and labsheets - particularly for the hardware component). It would be good to either simplify this component in the lecture slides, or make it clearer which are core concepts.

[6] I felt that pointers and structures, link list topics should be introduced well in advanced in week 3 or 4 and more focus and more practiced should be done on those things before moving to File I/o and directory and other topics where we need to use pointers, struct etc.

Also, I feel as a professor should write in pre-requisite that C topics should be well known to you before you start the course. If I knew that those things were needed too much, I might have started studying C more frequently. My impression was that I will be learning C in detail (each topic in detail).

Also, I felt more examples or some resource for examples should be increased so concepts get clear, more the people practice.

I felt till mid-sem the unit was going little slow but then suddenly it started to take speed and I was lagging behind before week 13 by around 2-3 weeks because of my other units clashes as well. So I feel if there is some pace to start from week 3 and students are advised that you sho

[7] I thought the assignments were cleverly constructed in that they encouraged you to learn how to use the C language while implementing and learning about system concepts. However, the first assignment was very time consuming and involved many sleepless nights; the second assignment required a much more reasonable amount of time.

[8] Chris is a great lecturer

[9] Would prefer to have a teacher that is able to help with issues.

[10] I enjoyed taking this unit as it really uncovers the wraps behind what actually happens in a modern PC architecture.

One area where I think this unit can be improved is in assessment. The assessments that we are given are often sensitive to changes in output or formatting due to the use of automated graders. It needs to be made clear on every assessment that automated graders will be used and example output for certain sets of input should be provided. It should also be clear exactly what criteria the automated graders will be marking on. In particular, test cases which are representative of the marking criteria would be very helpful in resolving any ambiguity.

These test cases need to be provided upfront and not posted on the assessment page or help forum in the days leading up to the due date for the assessment. This is to avoid students wasting time on trivial formatting / and or layout matters which would cause them to fail the assessment even if they had substantively comp

[11] Great unit, great lecturer. Challenging but not to hard, even though it's probably a tough unit to get the balance right.

[12] This unit should use rust instead becuase it's more geeky and that's all that really matters.

[13] I thoroughly enjoyed this unit.

Material was accessible easily without having to 'dig' through and locate the topic required.

The project tasks were strongly linked to current topics being discussed in lectures. The CPU simulation in project 1 - though frustrating - was, in retrospect, an excellent task that helped solidify my understanding of the model. The subsequent tutorial on pointers and the modification of project 1 to use pointers was a good way of showing their use.

The effort made by the UC in preparation for the lectures is evident in their use of live demonstrations. The demonstrations were immensely helpful in aiding my understanding of programming concepts. This comment extends to the 'live programming' tutorials where step-by-step examples of how to tackle a problem were shown.

These tutorials were extremely valuable - by watching someone else walk through 'best practice' and the relevant logic involved, I found it easier to apply the same concepts to the lab

[14] Very enthusiastic lecturer copped a lot of abuse from weird students for no reason. Very smart, interesting and clearly layed out course

[15] Too hard to learn for the beginner

[16] The exam has hard plz scale it up by +50% so I can pass. Pete Whittaker is a lad 420 swag swag money bitches

[17] I found that the content was covered a little to fast for me to keep up which made me feel behind a lot of the time. I had to go through a lot of the lecture slides myself after the lectures, and read and work things out for myself.

[18] The resources for this unit were well structured.

[19] The only area of improvement I feel is ensuring that all the printed lecture material is covered adequately in the lectures. Often time ran out and some slides weren't covered, and sometimes some explanations of elements of code samples were glossed over. On the other hand, I think the effort the lecturer went to in order to rapidly respond to so many student queries over the course of the unit is highly commendable.

[20] Very hard unit, definitely not for someone with no history of programming. As I was such a

novice I found it very difficult to keep up with the pace of the unit. There should be prerequisites for this unit. It is beyond me why there isn't. It also seem to be a fundamental unit for electrical engineers as coding in C is required for digital embedded systems. Units like this make me question whether UWA want there students to succeed.

Chris was a good lecturer very passionate and knowledgeable on the subjects he was teaching. This was great for the students who had good knowledge of the unit. However for novices such as myself it was difficult to ask a question without getting a sarcastic answer which is not helpful. I understand this is a level two unit which is designed to be more difficult but for a lot of students it is also an introduction to using the C language. The structure just baffles me.

[21] This was definitely the best unit I've taken in my computer science degree at UWA. I was impressed by how well this unit was organised and it's clear that Chris really cares about providing a fair and honest service to his students. While the unit is challenging (which you would expect from a second-year university computer science unit), it is fair, so I think Chris has definitely struck a good balance from this perspective.

[22] Overall the unit was very well run, the projects were a very large time commitment which was expected when entering the course, however I believe due that the projects may have impacted the other units I was taking this semester.
However I did believe that this unit is very well run and was a good experience.

[23] Assignments too hard by about 10-15%. Need gratuitous use of stackoverflow.com or to memorise textbook.

A bit too much jargon used ad hoc in lectures.