

DR ANDREW COLES

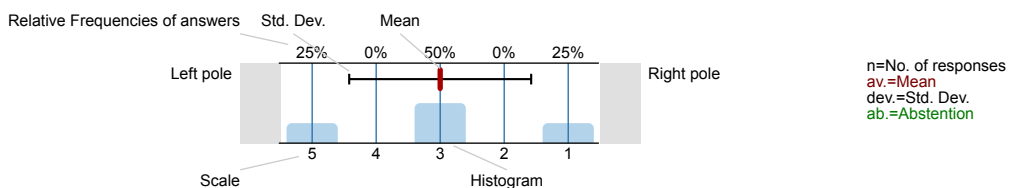
Practical Experiences of Programming (5CCS2PEP 2018/9 SEM1 000001)
 No. of responses = 46



Survey Results

Legend

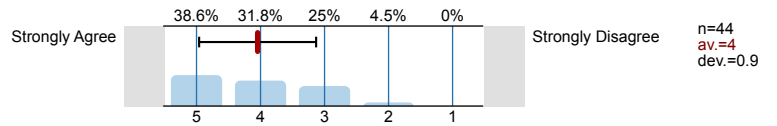
Question text



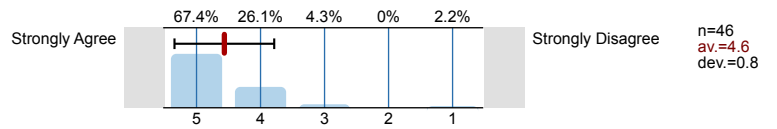
1. Module Questions

1.1) This module was intellectually stimulating		Strongly Disagree	n=46 av.=4.5 dev.=0.6
1.2) This module has provided me with opportunities to explore ideas or concepts in depth		Strongly Disagree	n=46 av.=4.5 dev.=0.8
1.3) The criteria used in marking for this module have been made clear in advance		Strongly Disagree	n=46 av.=4.2 dev.=1
1.4) Feedback on my work for this module so far has been received within the published timeframe		Strongly Disagree	n=45 av.=3.6 dev.=1
1.5) I have received helpful and informative feedback on my work within this module so far		Strongly Disagree	n=45 av.=3.4 dev.=1.2
1.6) This module has been well organised		Strongly Disagree	n=44 av.=4.1 dev.=1
1.7) Learning materials (e.g. handbooks, study guides, teaching materials and online content) for this module have effectively supported my learning		Strongly Disagree	n=46 av.=3.8 dev.=1
1.8) I have received good study advice and support when I have needed it		Strongly Disagree	n=46 av.=3.7 dev.=1.1
1.9) I have felt included in this module through having been encouraged to ask questions and participate in discussions		Strongly Disagree	n=46 av.=4.1 dev.=0.9

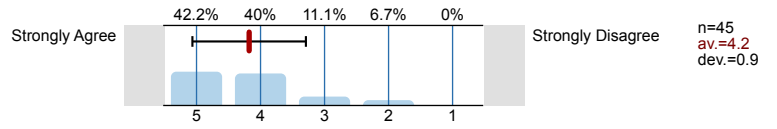
1.10) Staff value students' views and opinions about this module



1.11) This module has helped me develop knowledge and skills which will be of use to me in the future



1.12) Overall, I am satisfied with the quality of this module

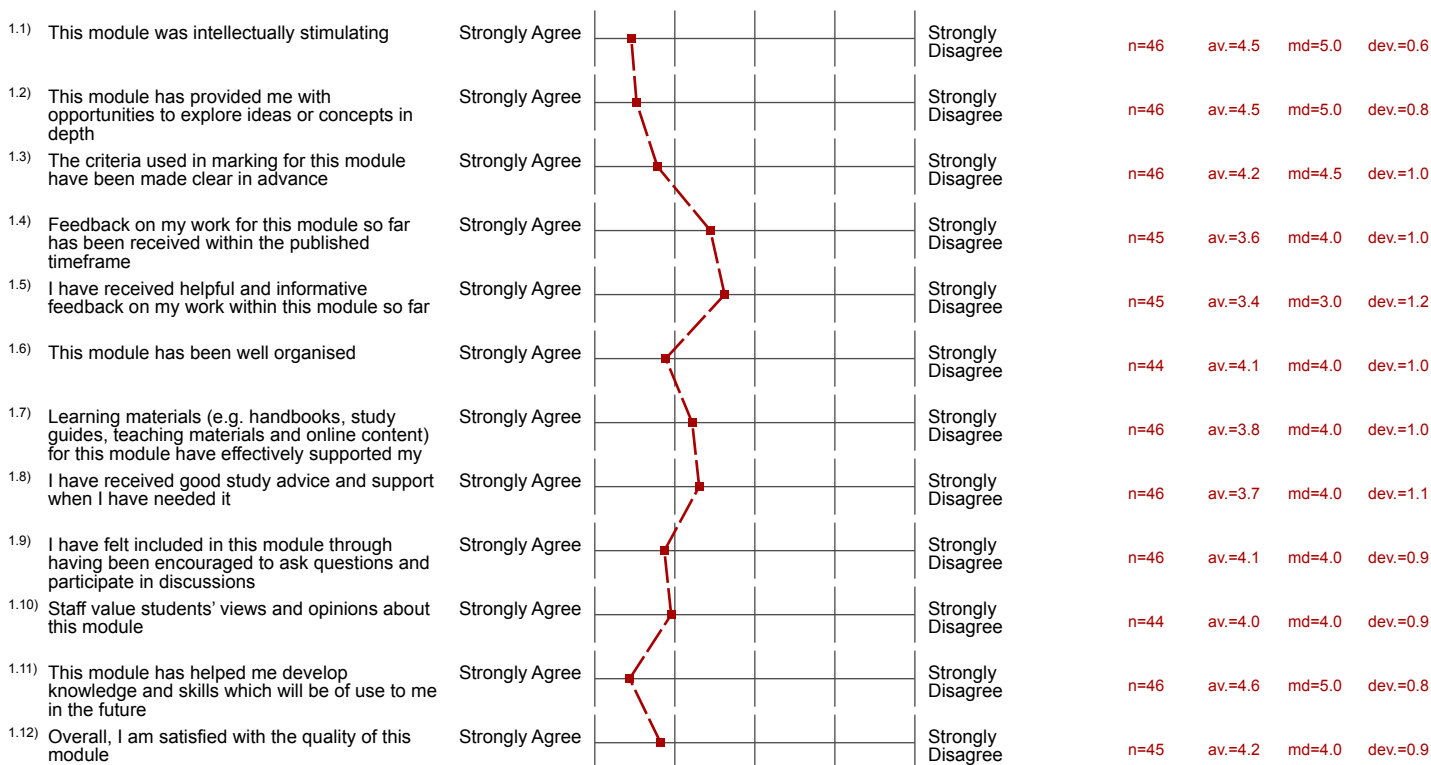


Profile

Subunit: **Informatics**
 Name of the instructor: **DR ANDREW COLES**
 Name of the course: **Practical Experiences of Programming (5CCS2PEP 2018/9 SEM1 000001)**
 (Name of the survey)

Values used in the profile line: Mean

1. Module Questions



Comments Report

1. Module Questions

- 1.13) What has been the most positive aspect of this module for you, and if you could recommend one improvement to the Module Organiser what would it be?
- - Quick testing
 - - Christian's testing should be similar to Andrew's (a file for test cases)
 - .
 - Andrew is an excellent lecturer and provided useful advice when needed.
 - Coursework pace extremely quick but that has lead to some very fast learning and adaptation to the pace and workload of second year has been made easier to have lots of smaller courseworks due regularly.
 - During the start of each new language, I regularly found myself investing 10+ hours into the questions for that week's work, and I feel this could have been avoided had we been given a 2-3 hour crash course to watch in order to learn the basics. Others regularly colluded on work and used last year's solutions to easily achieve perfect grades, meaning people who invested the time to do it ourselves and got 4 or 5 are being overshadowed by those who simply went and copied from someone else's work
 - Good lectures, interesting assignments
 - I have learn't a lot from this module, the content was challenging enough. I'd recommend improving the TA's since some of them were unable to answer questions. Also human feedback on the code would be helpful despite being expensive.
 - I really liked the C++ part of this module. Compared to Scala, it was more intellectually stimulating and I really enjoyed the challenges for each week. To improve, it would be great if the readme file for the scala projects had the actual description of the task, Like in C++ tasks.
 - Improve the testing of submitted coursework by adding a time stamp to all feedback issues (it can get confusing when multiple commits were pushed and one has to decide which feedback relates to the latest version of the code). Improve testing to be more specific, especially with the Scala assignments.
 - Learning the basics of a mainstream programming language, C++. Whilst I am not a fan of Scala, it was also useful to know how functional programming languages work. A HUGE improvement, mainly related to Scala lectures, is making the lectures 2 hours (and if there's too much content to cover, split it in 2hr and 1hr) because most people can't focus on programming for 3hrs. I always lost focus after around 1 and a half hours of watching someone else code.
 - Most enjoyable module, C++ was really fun
 - Most positive aspect of this module in my opinion are the resources available on Keats like the skills and tools assignments and video lectures. However, I think the time frame given to finish the lab videos which take 5 hours should be given more time to do or the work load spilt up differently.
 - Really enjoyed the opportunity to learn two programming languages that are so different from each other because it allowed me to find my strengths. The lecturers were both very good and kept me interested. the feedback for the advanced parts from c++ have been a little bit slow but understandable.
 - Release c++ advanced task marks please?
 - Sorry, many improvements: Better TAs that can actually help weak coders, less difficult ending assignments and advanced parts, stop giving such low marks for completing the first half of the assignments.
 - The issue is with my mixed reviews is referring to different lecturers- I felt one lecturer (Dr Coles) was significantly more helpful than the other. Additionally the assessment in the module was flawed as the marking system was often faulty/slow
 - The lectures have been great. One recommendation would be harder assignments, but with more time given to complete them.
 - The module provides opportunities to learn a lot of new concepts. However, the volume of work is overbearing. The feedback is minimal to none and there are no provided answers.
 - The module was perfectly organised and the information covered was both interesting and relevant. Exercises were intellectually stimulating and the solution was not straightforward, which, I think, makes this module very interesting. The only improvement could be the provision of supplementary materials for the course which would be relevant and get into more details about each programming language.
 - The module was well organized and taught. In my opinion there should not be any 9 am 3 hour lectures as it is hard to stay focused throughout the whole period of time.
 - The most positive aspect is that there has been a lot to learn (though others may argue this is negative). If there is any improvement to

be made I would suggest a gentler introduction to C++, it felt a little like being thrown into the deep end. Perhaps more focus on pointers, specifically allowing for one `unique_ptr` to point to a deep copy of what a different `unique_ptr` points to.

- The scala section is taught very well and workloads are reasonable. The c++ section was not taught as well and the workload given was too much to complete in a week.
- Weekly assignments were okay... up until the point where you have to do Scala, front end, and Java at the same time. But to be fair, the C++ part I loved, even though it was quite a hard grind sometimes. I can say that I laughed of stress and cried of breakdowns with this module and I totally recommend it for students who don't have jobs (y)
- Would have been useful to receive model solutions for coursework assignments.
- automated testing scripts were great, in the future I would give a concise deadline for when marks will be released