

## DR CHRISTIAN URBAN

Practical Experiences of Programming (5CCS2PEP 2023/4 SEM1 000001) (5CCS2PEP-2023/4-SEM1-000001)

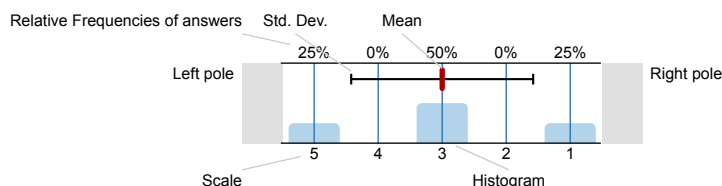
No. of responses = 100



## Survey Results

## Legend

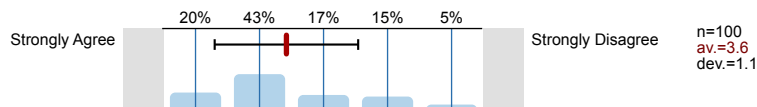
Question text



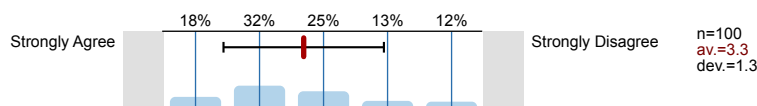
n=No. of responses  
 av.=Mean  
 dev.=Std. Dev.  
 ab.=Abstention

## 1. Senir Dinar - Lecturer Questions

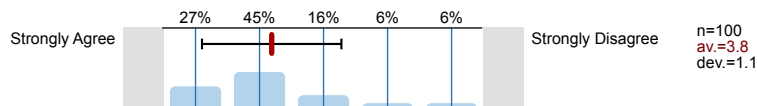
- 1.1) The lecturer has been good at explaining the subject



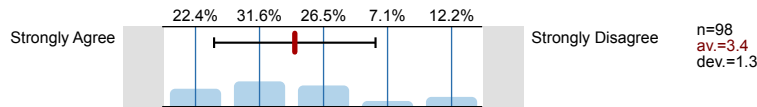
- 1.2) The lecturer has made the subject interesting



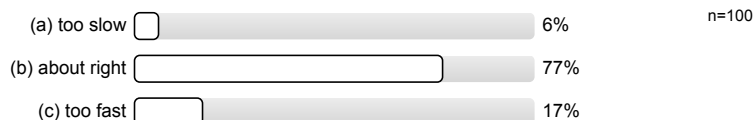
- 1.3) The lecturer has been well prepared for their classes



- 1.4) The lecturer cares about my learning experience

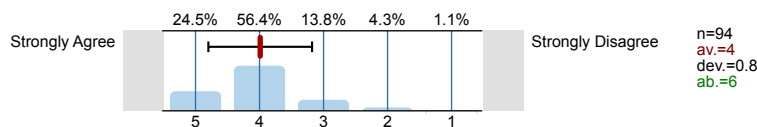


- 1.5) The pace of the lectures/seminars was

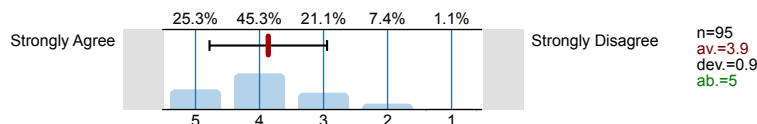


## 2. CHRISTIAN URBAN - Lecturer Questions

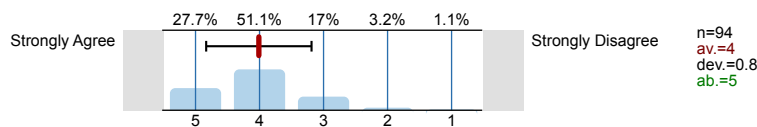
- 2.1) The lecturer has been good at explaining the subject



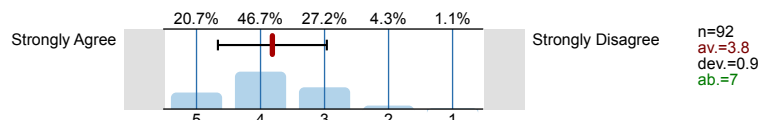
- 2.2) The lecturer has made the subject interesting



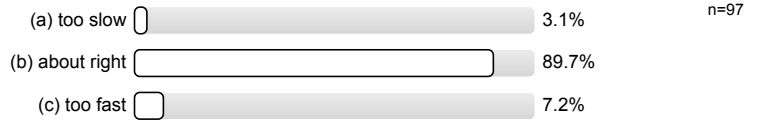
- 2.3) The lecturer has been well prepared for their classes



2.4) The lecturer cares about my learning experience

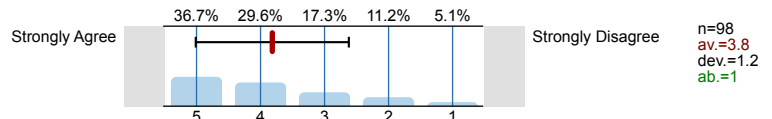


2.5) The pace of the lectures/seminars was

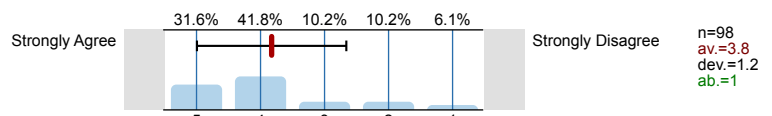


## 3. Practical Experiences of Programming (CORE) - Module Questions

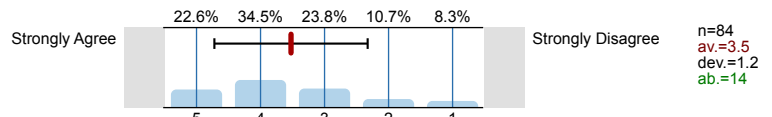
3.1) This module was intellectually stimulating



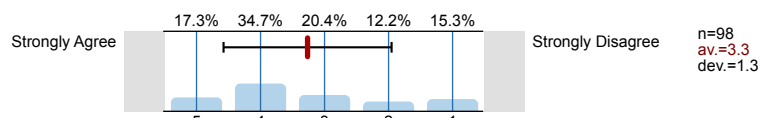
3.2) The criteria used in assessment for this module have been made clear in advance



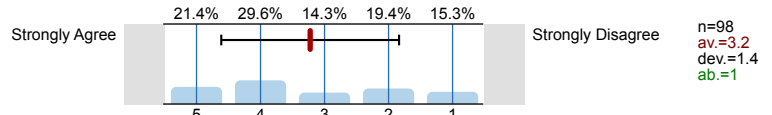
3.3) The written/verbal feedback I have received has been helpful



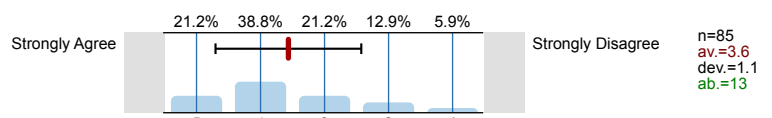
3.4) This module has been well organised



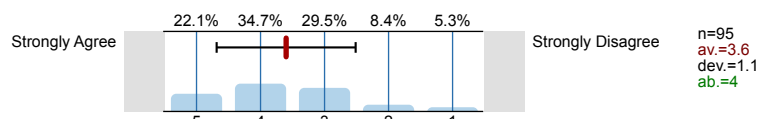
3.5) Learning materials (e.g. handbooks, study guides, teaching materials and online content) for this module have effectively supported my learning



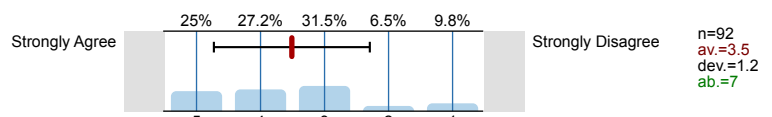
3.6) I have received helpful study advice and support when I have asked for it



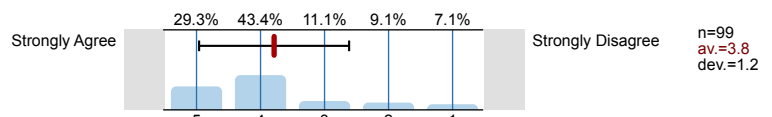
3.7) I have felt included in this module through having been encouraged to ask questions and/or participate in discussions



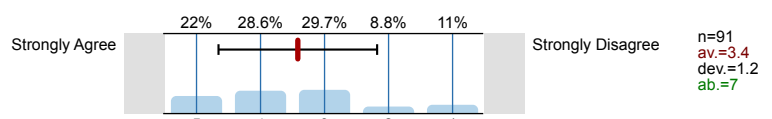
3.8) Staff value my views and perspectives in this module



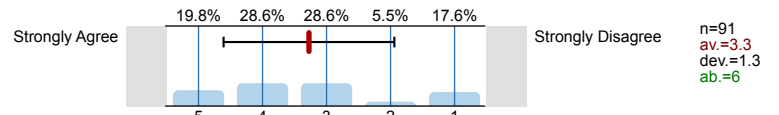
3.9) This module has helped to broaden my knowledge and/or skill set



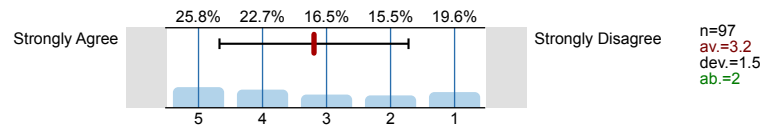
3.10) I feel part of a community on this module



3.11) The design and approach of the module made me feel included



3.12) Overall, I am satisfied with this module



# Profile





Subunit: Informatics

Responsible for modules: DR CHRISTIAN URBAN





Name of the course: Practical Experiences of Programming (5CCS2PEP 2023/4 SEM1 000001)  
(Name of the survey)

Values used in the profile line: Mean












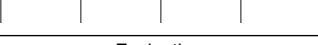
## 1. Senir Dinar - Lecturer Questions

1.1) The lecturer has been good at explaining the subject	Strongly Agree		Strongly Disagree	n=100	av.=3.6	md=4	dev.=1.1
1.2) The lecturer has made the subject interesting	Strongly Agree		Strongly Disagree	n=100	av.=3.3	md=3.5	dev.=1.3
1.3) The lecturer has been well prepared for their classes	Strongly Agree		Strongly Disagree	n=100	av.=3.8	md=4	dev.=1.1
1.4) The lecturer cares about my learning experience	Strongly Agree		Strongly Disagree	n=98	av.=3.4	md=4	dev.=1.3

## 2. CHRISTIAN URBAN - Lecturer Questions

2.1) The lecturer has been good at explaining the subject	Strongly Agree		Strongly Disagree	n=94	av.=4	md=4	dev.=0.8
2.2) The lecturer has made the subject interesting	Strongly Agree		Strongly Disagree	n=95	av.=3.9	md=4	dev.=0.9
2.3) The lecturer has been well prepared for their classes	Strongly Agree		Strongly Disagree	n=94	av.=4	md=4	dev.=0.8
2.4) The lecturer cares about my learning experience	Strongly Agree		Strongly Disagree	n=92	av.=3.8	md=4	dev.=0.9

## 3. Practical Experiences of Programming (CORE) - Module Questions

3.1) This module was intellectually stimulating	Strongly Agree		Strongly Disagree	n=98	av.=3.8	md=4	dev.=1.2
3.2) The criteria used in assessment for this module have been made clear in advance	Strongly Agree		Strongly Disagree	n=98	av.=3.8	md=4	dev.=1.2
3.3) The written/verbal feedback I have received has been helpful	Strongly Agree		Strongly Disagree	n=84	av.=3.5	md=4	dev.=1.2
3.4) This module has been well organised	Strongly Agree		Strongly Disagree	n=98	av.=3.3	md=4	dev.=1.3
3.5) Learning materials (e.g. handbooks, study guides, teaching materials and online content) for this module have effectively supported my	Strongly Agree		Strongly Disagree	n=98	av.=3.2	md=4	dev.=1.4
3.6) I have received helpful study advice and support when I have asked for it	Strongly Agree		Strongly Disagree	n=85	av.=3.6	md=4	dev.=1.1
3.7) I have felt included in this module through having been encouraged to ask questions and/or participate in discussions	Strongly Agree		Strongly Disagree	n=95	av.=3.6	md=4	dev.=1.1
3.8) Staff value my views and perspectives in this module	Strongly Agree		Strongly Disagree	n=92	av.=3.5	md=4	dev.=1.2
3.9) This module has helped to broaden my knowledge and/or skill set	Strongly Agree		Strongly Disagree	n=99	av.=3.8	md=4	dev.=1.2
3.10) I feel part of a community on this module	Strongly Agree		Strongly Disagree	n=91	av.=3.4	md=4	dev.=1.2
3.11) The design and approach of the module made me feel included	Strongly Agree		Strongly Disagree	n=91	av.=3.3	md=3	dev.=1.3
3.12) Overall, I am satisfied with this module	Strongly Agree		Strongly Disagree	n=97	av.=3.2	md=3	dev.=1.5

# Comments Report

## 3. Practical Experiences of Programming (CORE) - Module Questions

- 3.13) 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?
- A positive aspect of this module has been the live teaching courses, and being able to learn about coding in the modules. A negative aspect which I think the module could be improved upon is the difficulty of the coursework tasks, they felt quite advanced and ahead of the content taught in the lectures.
  - As much as the code on the forum for the coursework was helpful for some, it was annoying that it only came out after some of use had already attempted and written a large portion of the solution in our own way, which made us less willing to delete all our code and restart with the one of the forum
  - Challenging module, having to go from never programmed in a specific language, to being proficient enough to solve the problems. Scala has been more interesting with immutability.
  - Conceptually, the module is great and is an extremely valuable experience. However, the marks distribution across the first coursework (C++) is absurdly distorted considering one question is worth 50% of the grade. A greater balance needs to be achieved here.
  - Despite enjoying the Scala learning experience, I must express my dissatisfaction with the C++ module. Where video material felt irrelevant, impeding practical application. The excessive time needed to juggle coursework and learning, compounded by the ill-timed introduction of C++ coursework, was frustrating and ineffective. The lack of alignment between video content and coursework added unnecessary complexity. The disjointed approach hindered my ability to focus and succeed, leading to a less-than-ideal educational experience. Your attention to these concerns is warranted, and I hope for adjustments that enhance the overall effectiveness of the C++ module. The only redeeming quality of the C++ lecture was the TA (Oscar) who took on the responsibility of coding C++ live which was greatly appreciated. My problem was definitely not with the C++ language I was very excited about the prospect of learning it but the module was very poorly constructed which resulted in a bad learning experience
  - Dr. Urban is amazing, very helpful, really great overall. The coursework matches the expectations, very good.  
Dr. Dinar has no regard for the learning experience at all. The coursework has nothing to do with what we were taught, nothing. It more of a DST coursework, and even then it's way too difficult. It's a mathematics issue, not a C++ one.  
We learnt nothing. The most unsuccessful teaching experience I have ever had.
  - Dr Christian Urban's lecture videos are hilarious and make it fun to learn Scala. I believe that Dr Senir Dinar could offer an 'intro to algorithms in C++' week because the entire coursework was just algorithms, not much else.
  - Engaging but think at least recursion should be taught in the video content for C++
  - Even though the assessment does seem disconnected to the content, in the case of the C++ coursework, I feel the assessment was very useful for real world work experience. Therefore, the only thing I would ask for is more examples of the questions that come up in the assessment.
  - For Scala, Mr Urban goes through the exact code he has already shown from his videos. I think this was because he recorded lectures during COVID, and has just allowed us to see those videos, but because of that, his in person lectures are less useful, as he goes through all the same problems he has shown in his videos.
  - Getting instant feedback on code has been a positive learning experience. I want to personally thank Dr Urban and Dr Dinar for providing such a wonderful course.  
I thoroughly enjoyed it.
  - Github test feedback is very useful
  - I appreciate the difficulty of the questions from the coursework and how they are aimed to push everyone out of their comfort zone and to learn new skills.
  - I don't think the first half of this module has been positive at all, the lecturer Senir was not very useful in teaching the programming language and the whole module was really stressful because of the amount of work we had to do. The C++ coursework was extremely difficult and most of it was NOT taught in the lectures and I had to get a C++ course to even get started with it. Even though Ruben was somewhat helpful the coursework was just not proportionally difficult to the lab sheets or the work we did in lectures.  
The coursework overall was extremely hard considering that we were learning the language for the first time and the attitude of the lecturer was not very helpful as he told us to not worry about passing every single test but that is what we were graded on which contradicted a lot in my opinion.  
Overall, I was so upset about the C++ coursework. (also removing one of the questions will NOT help with reducing the workload)
  - I feel like it is probably one of the most important modules I have taken so far. However it has caused me a lot of stress and anxiety especially surrounding coursework as the lecture videos don't feel like they help with it. Having one coursework be released whilst another still has to be done does add to the stress.
  - I feel like the module's point is to help us experience what it is like to program very challenging problems, it's not very realistic as in a realistic scenario we would have access to the internet, and other people's code and may collaborate. I feel like the point of the

module is taken away due to how the plagiarism and collusion rules are put into place

- I feel like there was a disconnect between the videos and the coursework - apart from pointers which helped for part 3. It would be nice to go over critical thinking/ways to attack a problem as content rather than c++ language itself as none of it helped when it came to coding in the coursework. Part 1 was especially hard and the answer only came to me when watching week 6 of FC2 about graphs and such. After trying so hard the mental toll of doing one part continuously to no avail is draining and demotivating. In hindsight I can say the coursework was not that bad but that's only after ive finished it and I faintly remember what I went through when I was doing it.
- I liked the fact we were introduced to new languages, the different perspectives but the coursework introduces concepts for algorithm which has not been much covered, so would love it if we were introduced to them
- I liked the labs and the coursework questions as they were interesting and fun to solve. If I were to make an improvement, it would just be to make the deadline clearer, as there were different deadlines in different places (submitted before 4 but resubmitted with comments for 1 between 4-5). I also enjoyed the demo videos as they helped me the most to understand the C++ content.
- I like the idea of the module, and I like the fact that we can take our own time on the coursework. I liked the challenge of learning a language i was unfamiliar with.

My biggest complaints would be the clarity of the tasks themselves and the learning material. I believe that the tasks could have had more direct requirements (at least for c++). Many unnecessary questions had to be asked about the tasks, i think many of the rules should be stated more clearly. Also, i think the lecture content, especially on the last week, should have been more focused on the actual tasks we were doing in our coursework. The week 5 learning content was about inheritance when we didnt have to use it at all for our coursework.

- I think the most positive aspect of this module is that we are being taught two programming languages, and it is the first module in the course in which we cover a functional language. However, it would be nice to make the module longer, possibly lasting the whole year, so as to alleviate the workload and also possibly allow for a more in-depth exploration of Scala and C++.
- It was challenging. Specially I was out of my comfort zone while I was doing C++ but this process made my skills developed and made my mental stability stronger. However, The idea of teaching new language in 5 weeks and expecting students to do coursework at the same time is kind of not realistic aspect. Specially for C++, structure and content of coursework might be more relevant what we learned during the lectures.
- I would like to point out that in my opinion, the marking criteria for the first coursework was unfair. The fact that part 4 takes up half of the marks for the whole assignment and the fact that getting a zero on it does not guarantee getting 50% on the assignment as whole is a bit extreme considering the fact that we learnt the language in only the of span of 4-5 weeks. My coding knowledge and experience I have obtained from first year is extremely incomparable to the external research and effort I have spent on this assignment. Also, the assignment instruction sheet was not very clear and a bit contradicting as it did not point out important information which I had to obtain from the discussion forum including allowing the use of specific external libraries and the allowed testing times. My overall experience doing this assignment was negative and it made me feel extremely mentally exhausted. The most positive aspect was the github automated testing.
- I would say it's definitely the most chaotic module out of the rest, C++ in particular. The requirements were unclear from the start of what we could or could not do, the courseworks begins before you have even started your first PEP lecture which is ridiculous. The difficulty of the coursework and the amount of time it consumes has single-handedly set me behind on other modules which I am struggling to catch up on which should not be the case. This coursework wasn't much of a test of your C++ knowledge but actually a LeetCode/Competitive programming coursework instead, which is highly unfair.
- Learning a core programming language such as c++ is essential in todays work place, and Scala is also a useful and cool skill to know. One improvement to recommend would be having more interactive labs; more personalised and focused towards an individual person and their progression as opposed to a general class of whomever turns up.
- Learning new programming languages and also increased focus on programming rather than theory
- Loved the intellectual challenges presented by the module. Solving such complex challenges would be the best way to engage, learn a new language practically. The KEATS videos on C++ were extremely helpful for a starting point. I felt that more than enough help was provided by both the module leaders and the TAs. I am however, quite dissapointed that the challenging aspect of the C++ part of the module has been declining over the years - would have loved to try and pushed my limits by solving more problems while being given a shorter deadline.
- Make the C++ coursework actually relevant to the lecture videos and not make it absurdly difficult
- Mastering time management, balancing study and coursework, was the highlight of this module for me.
- Please teach in person
- Positive: I liked the tasks in the c++ coursework. They challenged my skills in programming and improve my overall experience in coding. C++ part let me explore more about debugging tools and they were helpful in C++ CW
- Positive aspect - Lecturers were very helpful during office hours., especially Senir Dinar. He gave me support for the coursework when I was struggling and was incredibly stressed.  
Module improvement - If we could get guidance on how to actual complete the coursework. For example in C++ we were taught nothing about dynamic programming or backtracking and recursion. If we could've tackled similar problems in the actual lectures then it would have made it easier to apply the knowledge to the coursework. It was quite upsetting to know that we really didn't get much support with the coursework, we were just thrown into the deep end. We come to the lectures to learn and not to get stressed and overwhelmed. I know that many people got extremely stressed with the C++ coursework.
- Prof. Senir is a very charismatic teacher.  
But we were entirely unprepared for the complexity of the first coursework and the high weightage of the very first assignment we got has made it very hard to score well. We would ask that the scale coursework be made of a higher weightage so students get a chance

to recuperate their marks.

- Reduce the pace of the module so that students are not learning a new programming language while simultaneously trying to do the coursework and not understand what is going on.
- Scala part of module was fine, my grievances lie with the C++ part. The video material felt irrelevant, impeding practical application. The excessive time needed to juggle coursework and learning, compounded by the ill-timed introduction of C++ coursework, was frustrating and ineffective. The lack of alignment between video content and coursework added unnecessary complexity. The disjointed approach hindered my ability to focus and succeed, leading to a less-than-ideal educational experience. Your attention to these concerns is warranted, and I hope for adjustments that enhance the overall effectiveness of the C++ module. The only redeeming quality of the C++ lecture was the TA (Oscar) who took on the responsibility of coding C++ live which was greatly appreciated. My problem was definitely not with the C++ language but with the lecturers refusal to understand why the coursework was hard, and his insistence on using Ruben (prob a paid actor) to make the module seem good.
- That I managed in such a short period of time to learn how to code in a new demanding language like C++ and complete tasks that include a lot of complexity. However, I think for someone that has no prior experience to C++ the tasks for the coursework as well as the content can be sometimes more challenging than expected and as a result students should read and study harder in order to keep up with the pace of the module which of course is something positive but at the same time hard because there are other demanding modules running as well. What I would recommend as a whole in the module is that at some point more actual tips are given for performing the tasks in the coursework or at least make students see more complex implementations of the things said in the videos in order to be better prepared for the coursework.
- The biggest concern I have regarding this course was that I perceived the coursework on C++ as a test of problem solving, rather than a test on C++. As a result, I would argue that a student could manage to score a really high grade without watching a single KEATS video, by learning the language "on-the-job" (while completing the coursework). A solution could be to assign small weekly c++ KEATS exercises (similar to the ones in SEG) which are autograded by Github, and can be worth 10% of the module (10% each for the C++ and Scala parts). These problems can purely focus on improving the student's understanding on concepts of the language: the linked list question was by far the question in which I learned the most about C++ itself, so problems similar to those should be included here.
- the C++ course work was stressful instead of learning C++ I relayed on my knowledge in Java, the questions did not vary in difficulty
- The content of the module was quite challenging, but I became a better student and programmer as a result.
- The coursework is too difficult
- The module does provide challenging tasks to complete, which is very good. I would recommend to have a comprehensive requirements set for each task at start of coursework, and not have unclear sections. I would also suggest to have the lecture content more focused on algorithms and strategies and not syntax, since syntax can be picked up quickly given PPA last year, but strategies and algorithms are more useful in the current year and for future. That would be a significant improvement.
- The separation between the lectures and the assignment is a very serious problem especially for Cpp part.
- The weighting is really good because it gives people a chance to do well enough even if they do horrendously on another one. Having said that, the c++ coursework was way too demanding for us. I believe that the content we were taught does not directly translate to the coursework because I was constantly finding myself looking for answers elsewhere on the internet than the actual content we were taught.
- This is a great, polished, module for which I only have one major nitpick (Mostly about the C++ side, as I've spent more time on it): I think it would be helpful to have much more precise information about the possible inputs of a function. Are the words in the input string guaranteed to be separated by one space, or is it unspecified whitespace? Can I assume the string is printable ASCII, or ASCII at all? Can I assume the input is solvable? What's the upper bound for input size? (This was specified for CW2) Obviously it's possible to say "plan for the least amount of assumptions, and all the edge cases" but there's a certain point where it no longer stays in the spirit of the exercise, and I need to know where exactly and precisely it starts becoming redundant.
- This module allowed me to solve programming problems and made me think about programming in different ways. Improvement: it is difficult to learn a new programming language and be thrown into the coursework at the same time. I feel like the c++ coursework in particular was disproportionately harder than what has been taught in classes
- This module has been great at expanding my knowledge in terms of programming and new programming languages and im glad we learn them in uni rather than having to learn it by ourselves, but i do think the module is very fast paced together with the other modules. I feel like i would have experienced more c++ and produced more content if we had more time.
- this module has felt more painful than childbirth. especially cpp. its a nasty language in the first place, why make the questions so nasty? i appreciate everything that was done to help us through it but it didnt get rid of the pain. i like how scala structures you through the coursework though, cpp should definitely adopt that.
- This module is extremely fast paced, especially in the first few weeks, but I understand that this is necessary and I think that it is handled as well as possible. The coursework for c++ is difficult, but understandably so, and the support provided by the TAs online was extremely helpful. The tests for the coursework were also a huge help
- This module was very helpful, and I thoroughly enjoyed it due to the substantial amount of information provided. It gave me a broader perspective on coding. Senir made the LGTs easy to follow, so I didn't feel lost.  
  
I would recommend improvements in the grading of the coursework. It perhaps would be beneficial for the harder questions (puzzle) to carry slightly less weight. That way, I would have experienced less anxiety about whether I could pass the module or not.
- To prepare students for the C++ coursework better, for example introducing recursion and/or backtracking, because that is a big part of the coursework but wasn't even touched upon in the videos

- We are taught basic C++ and given difficult problem solving tasks for the coursework worth 50% whilst barely knowing the language. Students are paying for GPT-4 to cheat which is undetectable, and generative AI detection is just snake oil that gives false positives. I have spent 40 hours on the coursework and barely got anywhere but someone with GPT-4 will get an extremely high mark in a fraction of the time - leaving me to wonder why I even bother trying.
- While giving us the opportunity to learn new programming languages has been enjoyable, I do not think the module has achieved its learning outcomes. Scala's approach to the coursework has been more useful as it is easier to understand what I need to do to get marks and is more helpful with regards to algorithms and techniques you need to know to do the coursework whereas CPP coursework was a complete mess. The CPP coursework felt like all it tested was do I know the correct algorithm to solve the problem and if I didn't know it then I get the question wrong.