

Project Title: **2022W1 UBCO Instructor SEI Surveys**

Course Audience: **214**
Responses Received: **65**
Response Ratio: **30%**

Report Comments

Recommended Minimum Response Rates

Class Size	Recommended Minimum Response Rates based on 80% confidence & $\pm 10\%$ margin
< 10	75%
11 - 19	65%
20 - 34	55%
35 - 49	40%
50 - 74	35%
75 - 99	25%
100 - 149	20%
150 - 299	15%
300 - 499	10%
> 500	5%

Legend

N: Expected
n: Responded

Frequency Distribution

SD: Strongly Disagree
D: Disagree
N: Neutral
A: Agree
SA: Strongly Agree
N/A: Not applicable

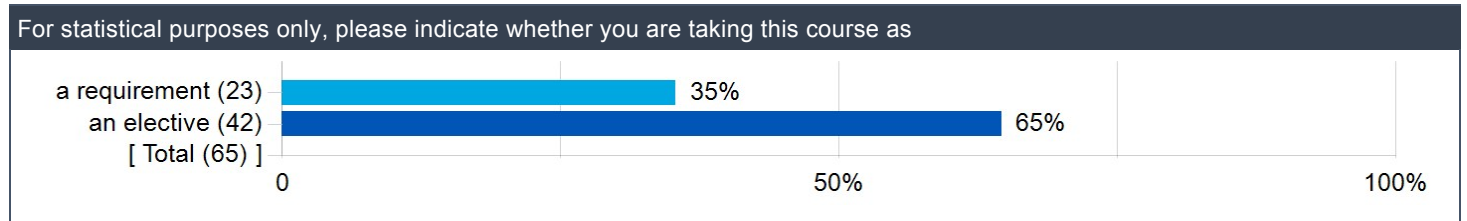
Statistics

IM: Interpolated Median

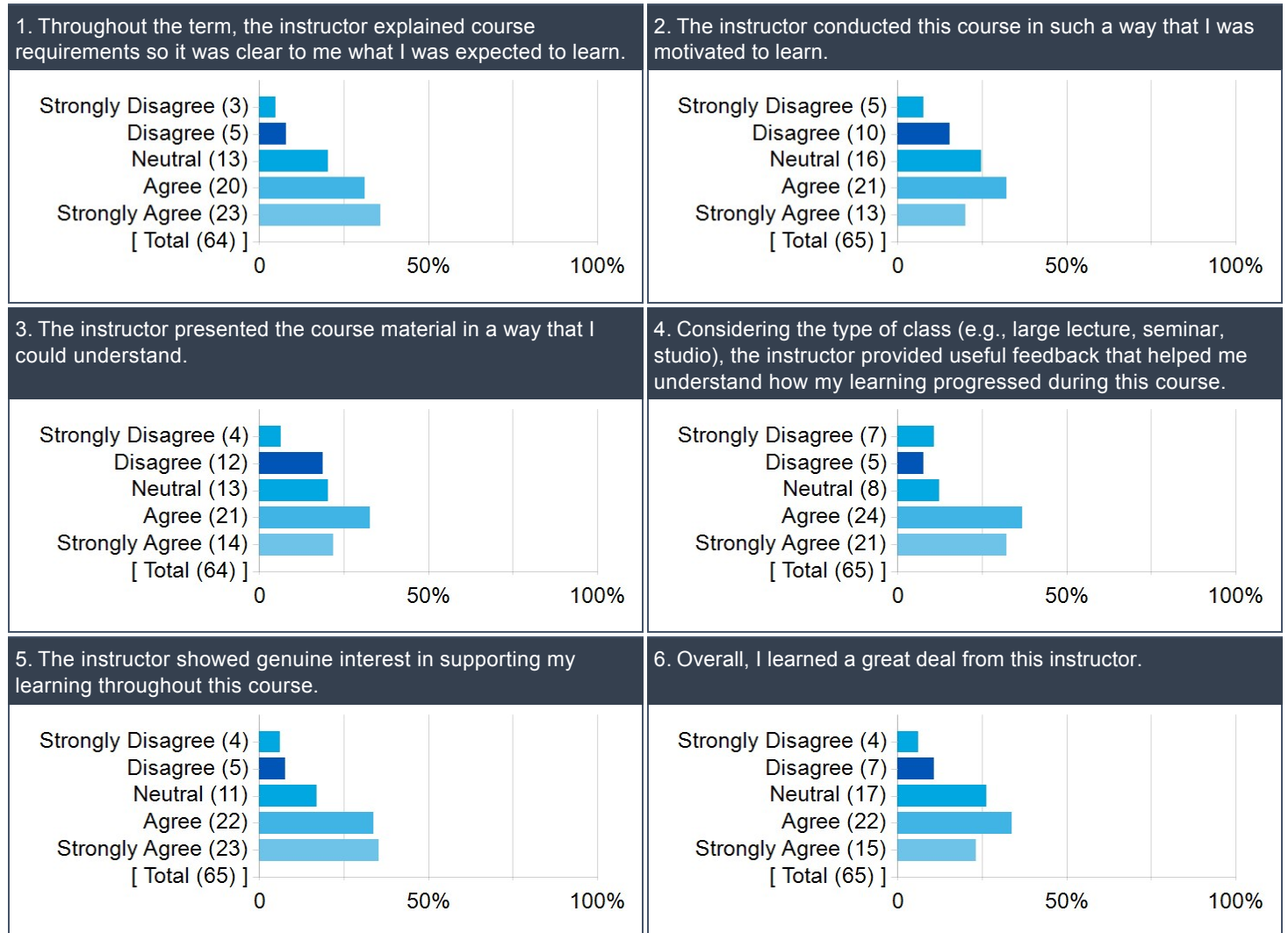
Creation Date: **Tuesday, January 10, 2023**

Detailed Results

For statistical purposes only, please indicate whether you are taking this course as



University Module Questions



UBCO Student Experience of Instruction

	N	n	SD	D	N	A	SA	IM	DI
Throughout the term, the instructor explained course requirements so it was clear to me what I was expected to learn.	214	64	3	5	13	20	23	4.1	0.6
The instructor conducted this course in such a way that I was motivated to learn.	214	65	5	10	16	21	13	3.6	0.7
The instructor presented the course material in a way that I could understand.	214	64	4	12	13	21	14	3.6	0.7
Considering the type of class (e.g., large lecture, seminar, studio), the instructor provided useful feedback that helped me understand how my learning progressed during this course.	214	65	7	5	8	24	21	4.0	0.7
The instructor showed genuine interest in supporting my learning throughout this course.	214	65	4	5	11	22	23	4.1	0.6
Overall, I learned a great deal from this instructor.	214	65	4	7	17	22	15	3.7	0.6

Question	%Favourable
Throughout the term, the instructor explained course requirements so it was clear to me what I was expected to learn.	67%
The instructor conducted this course in such a way that I was motivated to learn.	52%
The instructor presented the course material in a way that I could understand.	55%
Considering the type of class (e.g., large lecture, seminar, studio), the instructor provided useful feedback that helped me understand how my learning progressed during this course.	69%
The instructor showed genuine interest in supporting my learning throughout this course.	69%
Overall, I learned a great deal from this instructor.	57%

Open ended feedback

Do you have any suggestions for what the instructor could have done differently to further support your learning?

Comments
I dont mind the grading system in this class but at the end of the day it must be converted into a real grade for the University so it was a little ambiguous to how we were actually doing
He could have avoided the following: He made several critical mistakes in his course instructions which wasted approximately 10 hours of my time reading his useless instructions. Then he directed me to watch a useless video that he claimed was helpful. But he was irresponsible enough to not pay enough attention and resolve the problems on time. Assigning him to teach more than one course per-semester is a crime for me. The TA for my section was rude and offensive. Firas and the course's TAs admitted several times that this course had many mistakes.
Already does a good job.
I think one of your greatest weaknesses is assuming your students understand the content to the degree you do, and therefore you don't have to teach the material. The jump in difficulty between your course slides and the labs we had to do was unfair. Without TA instruction there is no way a majority of the class could have gotten through them. You need to provide a complete instruction on concepts. A line of code from begin to end that is successful to show a model to students. Too often you would explain part of a concept and then expect students to look up the answers for themselves, but without a model, shown multiple times from beginning to end you set your students up to fail. This is a beginner course and students should be treated like they've never coded before and given several opportunities to go through the process.
The university should provide the professor with more time for students hours. He has lots of students and I feel it could be beneficial for the professor and us students if we had more time to ask questions. The course is not simple and as this course is open to everyone, the starting point of people that don't have a background in COSC is lower and therefore might need more assistance.
I think that labs could have been easier and shorter to help keep up on other courses. There were so many assignments for this course that it took priority over other classes.
Maybe he could've toned down the volume of tasks. As university students, even minor exercises like "learning logs" take a toll on us. I feel like I wasn't able to gain the desired result from completing these logs. And missing some of them only harmed my grades and raised my anxiety.
More descriptive examples of the course content.
During the term we were told we could go to any of the labs, not just our designated slot. I wanted to use this, but as there's no where for me to find the time and PLACE of the other labs, I couldn't. I did try asking at least twice in the learning logs
I would have liked a progress report in this class.
Overall, my professor is fair and accurate in checking our labs and exercises. He is also supportive when students ask. However, during the lectures, more instructions on how to apply the contents to vs code should be clearer and given regularly since it can help students understand how it operates.q
I think that for a class with such complex concept like this one it needs thorough explanations so that students that are not in other courses in this field can grasp the concepts easier. Providing better notes that relate to the weekly labs that are required for us to do would be useful instead of having to rely on searching up functions and trying to teach them to our selves.
No suggestions.
For one, the content we learn in class and apply in the labs is entirely different. The things we learn in class barely helped when it came to the labs. It got to the point where we learned more from our TAs during labs than we did from moosvi during lecture. As well, the tests were very unorganized, a lot of questions had errors and asked questions we didnt cover in class. Overall, it was hard to understand what we had to know for tests, and test questions were trick-based rather than learning based. I couldnt imagine taking this course without any prior coding knowledge, because even haven completed programming courses, it was still very confusing as Moosvi does not provide adequate resources for the labs and expects us to find solutions online. Online solutions often confused students even more, especially if they didnt know what they were doing or had no prior knowledge. I took this course expecting it to be elective-level work, but because the labs were so confusing, it took up entirely too much of my time that I couldve used to do better in other classes.
taught more of the advanced html, javascript in class
Smaller class. With so many kids and so many question not all of them could be ever answered.
N/A
The learning labs were unnecessary and could've been shorter or completely omitted. In addition, the labs and content had a sharp increase in difficulty that was too difficult adjust to, especially for someone with little to no computer science experience.
gave the answers for questions asked in slides because i couldn't figure out how to do them and not having the answers made my

Comments
knowledge for the question incomplete
None
Teach more about how to do labs in classes
Make the course less lab and exam heavy so students can improve their grades
Making sure that everything is straightforward and easy to understand and find would be a great start. As this entire course is the only one that does not use a standard layout of canvas, I found it very difficult to find things when I needed them, and not everything was where I expected it to be (labs weren't in the weeks). Also when it came to labs, the instructions were never clear. Sometimes they included a lot of detail, other times they skipped over multiple steps and left me frustrated as to why I could not complete them.
As an older student (23) for this first year course this was my first experience with computer science and programming. I did not do any programming in my high school. I found in this course we were expected to do a lot of programming without being shown the basics first. The tutorials were not very helpful as the poor TA had too many questions to answer within the allotted time. I think that there should be a much greater emphasis on actually teaching how to program, for example, working through problems as a class, instead of expecting students to figure out how to do it on their own.
I seemed to struggle with coding questions on the test which I know is my own fault not his but, in the future, it would be nice if one coding question was not out of 10 points. I got a 53% on a test because I got one MC question wrong as well as the 10-point coding question even though I got all the other 13 questions right. Just a suggestion but then again that is my fault for not fully knowing how to do a coding question.
As there are no pre-requisite for this course, it can be hard for students who have no coding experience to understand the coding languages and instructions when not put into its most basic form. For example – in line 19, not beside the div.
His course was open book, which I believe is important for a Comp Sci class, this means not all the concepts of the course were covered in lectures. When taking tests a lot of the time he would stage questions like "We briefly touched on this in class, but don't worry it's open book so feel free to look up the concept". And then would proceed to make the answer confusing or "None of the above". It is hard to teach myself a concept within minutes to answer a test question when I don't even have the correct answer to go off of. Also, a lot of his tests had errors in them or "trick questions" and again, for a course that was open book and left you teaching yourself while taking a test having a trick question is difficult. I have no prior knowledge of Comp Sci.
I would say to be more strict and go more in depth with the course content.
One suggestion could be helping and guiding students on how to do labs properly. I believe that only giving the documentation for a student to do the lab by themselves or with a TA's help did not motivate me at all, because it felt like I had to do most of the work and reading myself.
Providing marks on the lab and the tests
Less bootstrap. I personally have a difficult time coding in general. However, when we're importing all sorts of code from 3rd party website it makes trying to code and understand the code significantly more challenging.
Have TA or yourself more available to student Mark assignments quicker Extra practice or video to build understanding Labs
Nope he was perfect
Dr Moosvi is so focused on making the learning experience good, getting feedback from students, and explaining the rationale behind his grading methods that he forgets to actually teach. His lectures were fluent enough, but the overall learning experience of the course ended up being worse because of how much emphasis he put on it.
I do wish Dr Moosvi would connect the labs and the lecture content more. He taught us pretty much nothing about Bootstrap, and yet we had to spend hours and hours of lab time on it.
Make the class content more related to the labs. One week in class we're learning about privacy and security, but in our labs, we have to make a GPA calculator. We are given no guidance whatsoever. Lots of people in this course have no coding experience whatsoever and things are poorly taught and explained to us for the labs. The labs are also worth 50% of our grades, so it's crucial to do good in the labs, yet there's very little support other than going to TAs to ask for assistance, but they will only ever give vague hints. If you have no coding experience, labs will take a huge amount of time to do.
–be more timely with feedback
I wish there could have been more examples gone through in class of different concepts because it feels like every concept was briefly brushed over but not in depth for us to truly understand it well. There were also some concepts that the labs required but wasn't really gone over during the lecture – ie. how to incorporate HTML and Javascript together, or what information about key functions and how they work.
– Certain materials (especially loops, array, conditionals if/else, function) should be given more attention such as clearer examples and instructions because it is very difficult to catch up if one topic is not 100% fully comprehended.

Comments
As someone who is new to computer science, I would have found it useful to see examples done in lectures. In lectures, when the lecture slides were being read, I was often confused. I think it would be helpful if the instructor didn't only read the examples on the slides, but also walked through the examples step by step so students can see how the code fits together. This would also help first time coders (like me) to see the 'coding thinking process' in action.

Please identify what you consider to be the strengths of this course.

Comments
I believe this was a great course if you are new to computer science, I am not so it was all stuff I had already learned but was nice as a review nonetheless
consistent assignments and tests that are never too much to deal with, all requirements and expectations explained as clearly as they could possibly have been
TA's were great.
The labs were definitely helpful being able to ask questions and having a smaller class helped. The fact that the professor let us do the bonus tests to see what we didn't understand go back review and learn from them.
This course had help provided when needed and lots of interaction tools.
Pretty straightforward.
Good labs. That's where I learnt the most.
The web site we do the assignments on works. Tasks included what we needed to download.
I liked learning Javascript and HTML
The contents of the lectures are understandable and easy to apply to real life, such as vs code. The lab instructions are easy to understand and very helpful in completing the labs.
The grading system was very helpful. The fact that the lectures were both in class and recorded was also very nice
The opportunity to be able to resubmit labs that you wish to receive a better score or prove your understanding.
This course taught different coding languages which will be useful in the future.
The labs are interesting.
cool way to learn the makings of a website
fast to understand the new things.
Labs and being able to join as many as we could for help
Being able to apply what we learned immediately after or during class
The first few chapters that were more elaborate
This course helps with some basic coding and mostly website development in HTML using bootstrap.
<ul style="list-style-type: none"> – easy to talk to for help –most understanding prof I've had – gave us an option to do bonus tests – gave us opportunities to improve our labs after submission – gave extensions for assignments if we explained how, it would help us and our plan of action.
The course allows for students to apply their knowledge from the lectures into their labs.
The course truly is for beginners. For example we had a lab on how to use Word, Excel, and Powerpoint.
The strengths of this course is definitely the ability to not be overly stressed about this course, allowing me to be able to focus more on the more demanding courses.
No mid-terms, open book on tests, bonus-tests, re-submission on labs, and 48-hour grace period. I did not have to memorize anything and felt like it motivated me to learn and reflect on my progress of the course.
The content
Resources mentioned above.
TA were helpful
Learning html
It is cool to get some early exposure to web development stuff. I enjoyed getting to see my code applied in the front end
It starts off easy, then you realize the professor can't teach and you're screwed
–Lots of opportunities to improve

Comments
–bonus tests are very generous –labs are engaging (other than lab 4)
The course allows for an extensive introduction to a lot of different concepts.
– Weekly tests instead of midterms which are extremely helpful for materials review and understanding – Flexibility for due dates
Openness to improvement and great an environment to build up your skills.
Labs show a good variety of skills to learn and use
amazing prof
Being able to ask questions through Ed Discussion was nice. I also appreciated having the bonus tests to try and improve my mark on tests.

Please provide suggestions on how this course might be improved.

Comments
Be responsible. Test the course content before experimenting with it on hundreds of students. I will find out who reads these surveys and question them about their incompetence because I wrote these surveys before and similar problems keep repeating.
Lab 4 was a bit much and could maybe be shortened or something but otherwise it's already good.
Lab difficulty needs to be brought way down, And course concepts need to be fully explained. Also, enough with the last minute changes to tests to make them harder.
Give the Professor more time for student hours. He has big classes and the concepts he teaches are not that simple people might need to ask questions but there might not be enough time.
Could have made the labs shorter this would have helped me stay on top of other courses as well some labs that were too hard could have hard more instructions.
Decrease volume of tasks. Also decrease course content. I felt overwhelmed with topics and this course was my introduction to com sci.
SL sessions/TA office hours
Have the lab times and locations listed on canvas, I'd recommend the unsyllabus
I did not enjoy learning bootstrap and using programs like gimp
Overall, my professor is fair and accurate in checking our labs and exercises. He is also supportive when students ask. However, during the lectures, more instructions on how to apply the contents to vs code should be clearer and given regularly since it can help students understand how it operates.q
Teach about topics that are used in the weekly labs to help reduce extra confusion students have trying to learn functions expected from us that were not gone over.
No suggestions.
1. Moosvi should teach the lab expectations during lectures 2. It should not be expected that students have prior knowledge of coding to be able to complete the course, as it has no prerequisites. 3. Moosvi should communicate and listen to our TAs 4. Provide better tests that are aimed to test knowledge on subjects covered in class and in labs 5. Moosvi should welcome criticism. Learning logs felt like a waste of time, and it felt like he didnt even look at the feedback provided in them
more focused on making website than having irrelated sub topics
More office hours and opportunity to practice
N/A
Have a more gradual increase in difficulty based on the curriculum from last year.
Labs could be done as a tutorial and not just as questions. For those who aren't familiar with the content just reading the instructions on how to do code was difficult. Following along visually helps me and talking to my peers that are also at the same knowledge level on
Have more proactive TA's instead of TA's that did not have enough time for students who are struggling.
Could take more time explaining the basic coding. I feel like if I hadn't taken COSC 111 I would have been pretty lost since it felt like it went by so fast.

Comments
The course was very difficult for an introductory course. There should be more time spent on the basics of programming.
– coding question not being out of 10 points
The course should have less labs. The course description was misleading as it explained we would be working with Microsoft word and PowerPoint which we only had one lab out of 10.
I would suggest to make the class easier for people that have no coding experience or prior knowledge.
By the end of the course we got into some pretty advanced stuff. I think it covers too much in one semester. I started off feeling good, each lab and test adding on from the last concept, but then it jumped and got much more difficult out of nowhere.
Create actual deadlines.
One suggestion would be making space for a lab explanation, and preview in the lecture, just like the first lab, markdown. Learning to code in the lectures, that way more people will be engaged and encouraged to learn.
Providing marks for the test and labs
Less bootstrap. I personally have a difficult time coding in general. However, when we're importing all sorts of code from 3rd party website it makes trying to code and understand the code significantly more challenging.
Be more available to students
Nothing
Who is this course supposed to be for? And what is it about? It seems to me that there is a split focus: computer fluency, and intro to web development. Here's why that is a problem:
– It is too shallow for computer science students. A lot of the course is spent on very general concepts of computers, which is good, but it doesn't really spend enough time on the coding stuff for it to be a true web development course.
– it is too technical for NON computer science students. Why bother with JS, HTML and Bootstrap? There are more user-friendly website creating tools for your everyday Joe.
– we don't spend enough time on anything in this course for it be that useful. A quick mention of excel doesn't teach you how to effectively use spreadsheets, but it does take time away from properly explaining what an array is. Arguably, by staying more 'surface level' this course is actually HARDER – because students have to do their own (sometimes extensive) research for any of the course content to be practical.
– In summary this course needs to be updated.
Have more guidance in the labs, make the expectations of the labs clearer, and have the labs be more related to course content.
–be more clear about deadlines and course requirements –don't make learning logs as long –the alternate grading system sounds good in theory, but i would've preferred a regular grading system bc it is somewhat discouraging to lose 15% of your lab mark for a single mistake
The course could have gone a bit more in depth on some of the more important concepts by providing examples or practice quizzes, and also could have added a section on how to actually combine JavaScript and HTML together.
The content we learned was too narrow to cover in one course. They needed to explain each topic more.
– Clearer instructions on how to do the labs – Heavier topics such as programming and JAVA should be given more time and attention and labs based on these topics should be given more instructions.
The grading system for labs doesn't seem to make much sense since every assignment is still assigned a number grade.
There seems to be a lot of content packed into this course, which led to it moving very quickly not learning very much about each topic. I feel like some parts could be cut in order to focus more in depth on the more complex concepts
a lot more difficult than I thought it was going to be, maybe explaining in greater detail before the drop deadline what exactly is expected of us
Overall, I think I would have taken more away from this course if I saw more examples being done in lecture. As the examples are being done, students would be able to work through that examples with the instructor (on their own device). This would allow students to get practice, to observe the thinking process, and to understand how the code works/fits together.

Explanatory Note

Percent Favourable Rating

This is the percentage of respondents who rated the instructor a 4 or 5 (Agree or Strongly Agree).

Interpolated Median

The data collected for Student Experience of Instruction (SEI) are ordinal in nature, with a natural order (from 1 to 5). While the mean may be used as a measure of central tendency for such data, it is not an appropriate or accurate representation of SEI data (cf. Stark & Freishtat, 2014). The usual measure of central tendency for ordinal data is the median. As a result, we have been reporting the mean and the median for the last several years. After considerable thought and data modeling, we now believe that the interpolated median is the best representation of the data, since it takes the frequency distribution into account.

Consider the following example from 2015W, the two course sections have identical mean (3.8). However, the instructor in section 2 received 77% favourable (4-5) ratings, compared to 53% for the instructor in section 1. The Interpolated median values of (3.7 and 4.2), much better reflects the distribution of the scores above and below their respective median. Furthermore, the interpolated median is better correlated with percent favourable rating; such that an interpolated median of 3.5 on a Likert scale of 1 to 5, corresponds to 50% favourable rating.

Frequency Distribution

Response for University Module Item	Section 1	Section 2
5 = Strongly agree	5	5
4 = Agree	3	5
3 = Neither agree nor disagree	6	0
2 = Disagree	1	2
1 = Strongly disagree	0	1
Mean	3.8	3.8
Median	4.0	4.0
Interpolated Median	3.7	4.2
Percent favourable rating	53%	77%

Dispersion Index

The dispersion index is a measure of variability suitable for ordinal data (Rampichini, Grilli & Petrucci 2004). This dispersion index has values between zero and 1. A zero dispersion index indicates that all respondents in the section rated their experience of instruction the same. An index value of 1.0 is obtained when respondents are split evenly between the two extreme values (Strongly Disagree & Strongly Agree), a very rare occurrence. In SEI data at UBC, the index rarely exceeds 0.85, and mostly for surveys not meeting the minimum recommended response rate.

