

# Report from Teaching Sabbatical at the National University of Singapore

July 3rd–December 26th, 2019

*Tobias Wrigstad*

Associate Professor in Computer Science  
Distinguished University Teacher  
Uppsala University

with invaluable input from

*Ylva Wrigstad*

January 30, 2020

# Contents

<b>1 Introduction</b>	<b>1</b>
<b>2 Preparation and Planning</b>	<b>1</b>
2.1 Planning Trip (22/2–3/3 2019) . . . . .	1
<b>3 Tasks and Responsibilities</b>	<b>1</b>
3.1 Other Activities . . . . .	2
<b>4 Reflection on Differences Between Uppsala University and NUS</b>	<b>2</b>
4.1 Unearthing the Excellent Students . . . . .	2
4.2 Singaporean Top Students . . . . .	2
4.3 Faculty Attitudes . . . . .	3
4.4 Student Attitudes . . . . .	3
4.5 Attitudes to Teaching . . . . .	3
4.6 The Importance of Exams . . . . .	3
4.7 Small Class Teaching . . . . .	4
4.8 Student Career Development . . . . .	4
4.9 Use of Technology in Teaching . . . . .	4
4.10 Extra-Modular Credit . . . . .	4
<b>5 Comments on the STINT Teaching Sabbatical Grant</b>	<b>5</b>
<b>6 Living in Singapore</b>	<b>5</b>
<b>7 Going Forward</b>	<b>6</b>
7.1 TUFF Grant 2020 . . . . .	6
7.2 SigCSE Grant for Mastery Learning . . . . .	6
7.3 Future of SICP Book . . . . .	6
7.4 Head of Education . . . . .	6
7.5 Miscellaneous . . . . .	6
<b>8 Concluding Remarks</b>	<b>6</b>

## 1 Introduction

This is a report of my (or more correctly the entire family) sabbatical at the National University of Singapore (NUS) July–December 2019 at the School of computing.

## 2 Preparation and Planning

In late December, I was contacted by Martin Henz, associate professor at the School of Computing (SoC) at NUS. Martin presented me with options for various courses in which I could be involved, but it was quickly quite clear that the course that was the best fit for me, especially given my home department’s interest in large-scale teaching, was the full-semester course CS1101s, “Programming Methodology”.

### 2.1 Planning Trip (22/2–3/3 2019)

I exchanged around 30 emails with Martin before we climbed aboard the Finnair flight to Singapore for the planning meeting, during Week 9 (“Sportlovsveckan”, at least

in Stockholm where we live). In addition to discussing various parts of CS1101s, he introduced me to William Jobe, STINT fellow in 2018 who he had also hosted. As it turned out, William also worked on CS1101s, and judging from his report had a similar experience as I. For a full NUS+SoC treatment, I suggest reading his sabbatical report together with this one.

The planning week time was split between shopping for schools (my daughter was 7 years old at the time, and in Swedish 2nd grade) and talking to Martin. In that week, I gave two talks at SoC, one covering some of my recent research on programming languages, and another on my work on mastery learning. Sadly, Week 9 was recess week, so most senior faculty members were travelling.

Organising school visits in Singapore is quite simple and handled effectively online through various forms. There are many school options, and choosing between them is (in our experience) extremely hard, possibly less so for someone with experience from international, private schooling. For us, visiting the school and getting a tour was an indispensable part of the school selection.

From a Swedish perspective, Singaporean schools cost an arm and a leg, and navigating the various fee structures can be daunting. For one semester, we paid around 17k SGD not including additional fees for school trips, school uniforms, school lunches, etc.

## 3 Tasks and Responsibilities

I teamed up with Martin and Kok Lim Low as a co-teacher for CS1101s. Both Martin and Kok Lim were exceptionally good at making me feel like an equally valuable member of the academic staff even though I of course knew very little about NUS rules and regulations, and had never taught the course before. I was invited to participate in pretty much any activity in the course, and I tried a bit of everything including (but not limited to):

- Collaborating with Martin on the adaptation of the course book to the Source language used in the course
- Develop parts of the gamification elements of the course
- Vetting assignment (e.g., instructions, the tasks, automated tests)
- Quality assurance of the course’s 80 Teaching Assistants (TAs)
- Coaching TAs in need
- Teaching “reflection sessions” (45-minute lectures with around 20 students focussed on repeating practical aspects of the material from the lectures)
- Taking consultation from students
- Organising extra classes for students in problem solving
- Participating in weekly staff meetings

- Participating in CS1101s events
- Help with setting and correcting exams
- Give a guest lecture on the last chapter of the course book

The CS1101s course is *very intensive* for students. The course follows the classic computer science text “Structure and Interpretation of Computer Programming” (by Abelson and Sussman), but adapted to a more recent programming language. All courses run for the length of a semester and students take several courses in parallel – in the case of the freshmen enrolled in CS1101s, this number was typically five. At the IT department in Uppsala, students typically take no more than three courses in parallel. Clearly, CS1101s was one of the more demanding courses, as witnessed by the students’ feedback, but also not surprising since programming methodology is a core subject for computer science students.

As CS1101s had around 550 students and 80 TAs, it gave me the perfect opportunity to study large-scale teaching. In Uppsala, we give several introductory courses in programming with overlapping content, and some opportunities for combining them exist.

Some key take-aways from this experience of large-scale teaching:

- The huge number of students exacerbate problems which are normally small: any fringe situation or freak accident will happen; tiny instructional mistakes (e.g. unclear wording in an assignment) create enormous volumes of communication
- For the TA–students ratio to remain reasonable, the time spent interviewing TAs is quite considerable
- Time spent on quality assurance (including TA consistency) is considerable and warrants careful design to be cost-efficient
- Unless communication structure is carefully designed, or the teaching organisation is too flat, the amount of issues that “bubble up” to the academic staff is huge
- Teaching a class of 500 students is no more impersonal than teaching a class of 100 (for the teacher)

### 3.1 Other Activities

I attended a number of “recruitment talks” from junior researchers seeking employment at NUS. This was interesting, both to see the range and the quality of the applicants, and to hear the questions asked by faculty members. In my home university, recruitment is nowadays handled centrally, and the receiving department do not get to meet candidates until they have been given an offer. This of course helps avoid nepotism and promotes consistency across many departments, but puts extreme pressure on external experts to weed out “bad cases”.

I attended a course at CDTL, the Centre for Development of Teaching and Learning, about Rubric development. It

strongly reminded me of the learning labs both at KTH and Uppsala University – from furnishing and seating arrangements, to rhetoric and teaching style.

I attended the NUS Festival of Learning, a two-day event organised by the CDTL, with a wide range of faculty members talking about their experiences and running workshops. It strongly reminded me of the local TUR conferences held annually at my home faculty, both with respect to content and quality.

## 4 Reflection on Differences Between Uppsala University and NUS

### 4.1 Unearthing the Excellent Students

Soon after I arrived at SoC, a school-wide staff meeting was held during which the head of department proclaimed that 35% of all Singaporeans attend the university, the top 15% of these come to NUS go to NUS, and the top NUS students currently come to SoC, because of the late surge in popularity of computing. A recurring theme in CS1101s was the need to challenge students, and provide some really hard assignments and exam questions to “clearly separate the excellent students from the rest.” As grades are very important, and also fighting grade inflation, this was not so strange in hindsight, but the rhetoric was very different from what I was used to (despite having spent two years in the US as a postdoc).

### 4.2 Singaporean Top Students

I found the Singaporean students to be extremely hard-working, very polite, and very diligent. More or less the behaviour one would expect from “top students” (for some definition of that leaning towards the classical sense), but it was at times surreal to have an entire class of >500 top students.

However, it was clear that the students had developed techniques for getting good marks more than they had developed techniques for learning. (I would say the same holds for most Swedish students as well.) Many students approached the course by trying to reverse engineer results from old exams and apply rote learning to specific programming techniques and subsequently did not do very well in the many exams given throughout CS1101s. Being top students, they had very little experience in “failing” (read: not getting a top score) and this naturally caused additional stress and questions of belonging, leading to vicious circles.

Students would not complain (at least never to a professor), and were prone to self-blame if they could not handle the workload. I once asked a group of students who visited me regularly for consultation whether they thought the assignments were too hard. They replied that the assignments were not too hard – the problem was that they spent too much time doing them.

The latter is perhaps related to another observation – that students did not seem used to critical thinking. They threw

themselves at whatever task set for them by a professor and never seemed to reflect on (let alone question) the reason for doing this or that. Consequently, they also had no basic understanding of how to limit themselves. For example, an attempt at a cursory introduction of some complexity concepts through shallow coverage of the concepts sent the entire cohort down a rabbit hole of filling out the blanks themselves, instead of interpreting the teachers' behaviour as an indication of the intended level of mastery at that point in time.

As my daughter was enrolled in the 3rd year in Singaporean international school during the fall, I observed the vast difference between how pupils are motivated in Sweden (usually via pull-based methods) and in Singapore (typically via push or decree). Although I have way too little data to generalise, it is easy to believe that Singaporean students have been shaped by lower levels of education to be expected to simply do what they are told, no questions asked. This is in contrast to Swedish students, who, in my experience, are prone to ask "why", and will want to know where something fits in a grander scheme of things.

Naturally, students who pay a considerable fee for a top education are probably more inclined to want to "get the most out of their education", in comparison with Swedish students that typically perceive that their education is free and (therefore?) have few demands on teachers or quality of courses. Nevertheless, it was not clear to me that the approach taken by Singaporean students give better results than those obtained by the top Swedish students.

### 4.3 Faculty Attitudes

NUS has an interesting website set up by a faculty member called NUS whisperers where students can (optionally but generally anonymously) vent or address things ranging from courses to love interests.

A couple of weeks into the semester, a freshman student posted the following:

*CS1101s workload is too heavy for any freshie to handle. No doubt the module is quite cool, but people with no programming background will die in this module. There are so many concepts to handle at one go and the missions just keep uploading.*

The first reply came from the moderator:

*Yada yada why are you at NUS if you complain about a world class education*

The second reply came from a well-known faculty member:

*Quit complaining. [...]*

*If you cannot take it, please change major*

The latter reply got a large number of "likes".

I expect that a similar behaviour from faculty at a Swedish university would (and rightfully so) result in some disciplinary board action or equivalent.

### 4.4 Student Attitudes

While students were very polite, they could, interestingly enough, never keep quiet in a lecture situation (regardless of size), which struck me as off-character. One explanation offered from a distinguished educator from NTU (Singapore's number two university) is that the students' exposure to active learning causes them to immediately start discussing among themselves when they find something hard or interesting.

Students had a hard time initially with my designation. They were used to refer to professors as "prof", sometimes with a last name thrown in as an extra qualifier, but given that my official English title is Senior Lecturer (as we use British and not American titles in Uppsala), this provided a problem. I tried getting them to call me by my first name, but eventually they all called me "prof". I fondly recall one day in which I was called "Prof", "Doctor Tobias", "Tobias" and "Boss" all in one hour at the university.

With students being so eager to please, and quick to execute suggestions, it was important for me to change my behaviour towards the students as I quickly realised that an off-handed remark (e.g., "this is an interesting book") could have unexpected consequences ("I've read this book that you recommended").

### 4.5 Attitudes to Teaching

Although teaching is held in very high regard at my home department and university, I found that teaching was taken very seriously at NUS. For example, auscultation by other teachers happens at regular intervals and produces a report with helpful suggestions from the visiting teachers.

At the end of the year, all teachers at SoC who had taught freshman courses, gathered for the "marks meeting" which involved presenting each course quickly to each other, and discussing how well this year's instalment had worked, and showing the mark distribution of all students in the class, and suggesting a mapping from marks to final grade. This mapping was then motivated, and a decision was made at that meeting for how to map marks to grades. Notably, this meeting was led by the head of the department. Also notably, failing students were discussed and care was taken to check how failing students in one class were doing in other classes. I have never seen or heard of something similar in a Swedish university.

At the end of the semester, a "teachers' luncheon" was organised, which was comprised of four 30-minute presentations. This was a very well-attended event.

NUS also has a "Educator in Residence" programme which is a great way to attract high profile academics to disseminate their didactic and pedagogic results to NUS faculty.

### 4.6 The Importance of Exams

Students in CS1101s had no less than five exams, in addition to the around 20 graded assignments and 10–15 optional assignments aimed at "best students". Of these

five exams, one was a practical assessment held in computer labs, two were multiple-choice exams, and two were more classical programming exams with a combination of showing understanding of basic concepts by answering questions about valid but non-sensible programs (e.g. “what is printed when this program runs”) and solving small programming assignments.

The final exam is a requirement from NUS on its courses, and no amount of clever course design can remove the need to assign a large chunk of a course’s marks to a brief exam on a particular day, set well in advance. As someone who strongly question the validity of exams (especially since we seemingly never take the required time to verify that an exam actually tests what we want it to, and not e.g. a student’s ability to understand how the teacher thinks) I lobbied hard to change exams, and I feel that both Martin and Kok Lim understood and respected my input.

Nevertheless, it seems that final exams is a central part of all courses at NUS, and semesters are carefully planned to give students ample time to “study for exams” in a way that we now know do not promote deep learning.

#### 4.7 Small Class Teaching

NUS believes in a small number of students per instructor, and in particular in a small classroom setting. I noted that the student–TA ratio of CS1101s is roughly the same as the course I have taught in Uppsala for almost 10 years, IOOPM (about 8:1). However, whereas we typically use TAs as “flying goalkeepers” in a large lab setting (typically to help those who are stuck and/or for correction), CS1101s tries to create small, tightly-knit groups where one senior TA teaches and mentors a small group of about 8 students throughout the semester. This means that a student’s CS1101s experience is very much a product of who their TA is, but tight quality control by academic staff and feedback from students make this work well in practise. I am still very enticed about the idea, often touted by Martin, of creating communities of learners. And I saw TAs being very involved in the learning of “their” students, and reasoning very maturely about the development of their students’ skills and abilities. This would be interesting to try to replicate in a Swedish university setting.

#### 4.8 Student Career Development

NUS cares a lot about career development which starts already at the “Welcome Tea” for the incoming students. At this event, NUS career developers come on stage and start talking about making career plans, including planning to make a stop at the career office in the first year, etc.

During this event, NUS clubs (a club for programmers, one for pairing programmers with student entrepreneurs, one for gamers, etc.) market themselves to students. One such club gathers programmers that collaborate on developing software pro bono for non-profit organisations. Their selling angle is making students more eligible for internships which makes them more attractive to top companies like Google, Apple, and Grab.

SoC proudly displays a list of companies formed by SoC alumnus, and also touts a startup programme open to both student and faculty.

#### 4.9 Use of Technology in Teaching

SoC teachers seem to be very keen to develop their own software for a multitude of things.

- Their own programming language for teaching introductory programming to freshmen
- Their own online programming environment that can draw the same explanatory graphs as in the textbook, support collaborative editing, graphics, sound and video processing, etc. (Several such systems built at SoC!)
- Their own presentation system for more interactive presentations
- etc.

These systems are constructed both by faculty members on the research tracks and teaching tracks. Most (or all) students take tons of notes, typically by writing on handed-out slides on a tablet or a computer. *The Singaporean school system has definitely trained students in attending lectures in a way that Swedish universities seem to take for granted (with sad results).*

One clear advantage over what I have seen so far in Swedish education is a system for letting students learn by building software used in their education. For example, the Source Academy software used in CS1101s is built entirely by students supervised by Martin in the CP3108 individual project course. Apart from forcing these students to learn frameworks and technologies that fall outside of traditional computer science courses (for good reasons), the students get experience in deployment and responding to hundreds of users’ bug reports and requests. A great learning experience.

The teaching load of SoC professors also seems considerably lower than at the IT department at Uppsala University, which perhaps provides the necessary time and peace of mind to construct such software.

I did not observe any use of cheap parlour tricks like clickers etc.

Courses seem to heavily use active learning labs where students work together in smaller groups around a table for 5–8 people, each equipped with a screen that can either be hooked up to a laptop on the table, show a laptop from another table, or show material from the teacher. This was inspiring and worked extremely well to move away from traditional lecturing with students facing the teacher.

#### 4.10 Extra-Modular Credit

NUS uses a system where a student can earn credits for a future course during another. In CS1101s, students that take the optional assignments (called “quests” in CS1101s parlance) earn “experience points” which they can convert

into one modular credit (modular credits are the NUS equivalent of ECTS credits) in the course CS1010R, which is an “add-on” to CS1101s that adds a research component that enables students to acquire more in-depth understanding of subject. This strikes me as an excellent way to challenge advanced students and keep them from being bored, and being able to reward them for their extra work.

The computer science subject must deal with the fact that a non-trivial number of students each year come with considerable pre-existing knowledge of programming, that puts them leaps and bounds ahead of many of their fellow students who have none or very modest exposure to programming ahead of university education. This puts pressure on software developing courses in general and programming courses in particular to find ways to challenge such advanced students without raising the bar too high for students with little prior programming experience.

## 5 Comments on the STINT Teaching Sabbatical Grant

Other teaching-interested colleagues from around the world have expressed envy and awe when they have learned about my teaching sabbatical and the grant conditions. The phrase that every country should have such a grant scheme has been recurring.

## 6 Living in Singapore

For a STINT fellow going to Singapore, here are some hints which may prove helpful:

- Singapore is not a cash-less society and ATMs charge a fixed overhead for foreign withdrawals. Plan ahead and keep wads of cash in your safe at home.
- Bank transfers cause delays and inflate costs and are necessary for fees related to children, e.g. extra-curricular activities like swimming lessons.
- NUS provides excellent help with getting work passes, but will charge you around 250 SGD per dependent pass. This fee cannot be paid by credit card, and required in my case visiting several ATMs to obtain the necessary amount of cash.
- NUS/SoC administrators do not like to give negative answers to questions in emails – making actual phone calls proved very effective when certain questions were repeatedly ignored.
- Unless Sweden abolishes DST, time differences will change during the semester, so that if you set a meeting according to Swedish time, it will move forward one hour.
- Some hospitals (notably the NUS hospital) does not accept the Swedish Kammarkollegiet insurance *unless* you are admitted. Check with Falck security before you go anywhere to avoid suddenly facing

having to go somewhere else because of something as mundane as money.

- To a Swede, Singapore is extremely *hot*. This acute feeling passes after (in our case) about a month, after which it did not feel unreasonable to wear e.g., sneakers or clothes with slightly more cloth. After walking to the office every morning, a 15-20 minute walk including some self-imposed detours for my step count, the first thing I had to do at work each morning was to change shirts.

We had to extend our wardrobes with clothes that felt right for both climate and context. Clothes that you own that work well at +30°C with 90% humidity might be better suited for leisure than lectures.

- There is an infinite amount of things to do. Excursions in nature. Walks. Museums. Amusement parks. Everything we tried was top notch.
- City planning is very different from Swedish cities. Blocks of houses often contain huge food courts with cafés and dozens of restaurants. This is often impossible to see from the outside, so do explore!
- Pharmacies are only found in conjunction with hospitals.
- Commuter traffic is great. Well-connected, reliable and punctual. However, it does take very long time to get around because of the frequent stops. A Grab or Taxi is very cheap and reaches most things at a fraction of the time of the bus or metro, but is a lot less quaint and less environmentally friendly.

You can ride the free NUS bus all the way to the (amazing) botanical garden.

- Lunches and dinners are cheaper to buy at hawker centres than to cook yourself. Tons of options, and delicious. I will miss the Indian food at The Deck for the rest of my life. Youtiao, chicken rice, noodles. Explore!
- Accommodation at Kent Vale is simple, luxurious and comfortable. Stable wi-fi, decent cookware, and good AC. The two swimming pools. Kent Vale’s apartments include breakfast which is very convenient in the mornings as you will quickly learn the need to wipe the kitchen and dining area meticulously clean each day to avoid feeding the ants. All the staff at Kent Vale are nice and friendly.
- Singapore feels extremely safe. Everywhere (where we ventured) and at any time of day or night. (Walking home from the office at midnight was amazing for the nightly sounds, especially the birds.) Singaporeans use practise “choping” which translates into calling dibs on seats and tables at restaurants, by leaving some random item – from a pen to a cellphone or a purse.
- IKEA Singapore is great for treating home sickness (including ostbågar and pepparkakor och lite lakrits).

- Pro tip: organise visits from home and have them bring you stuff you lack, like more dark roasted coffee. On the subject of coffee, Singapore coffee is great, except that one must explicitly exclude certain supplements. Black coffee is “Kopi o kosong” and all milk is condensed and thus (too) sweet.
- Singaporean English can be extremely terse and can take a while to get used to. It has been explained to us as basically Chinese spoken with English words. For example, the question “What should we do if a student goes to the wrong exam hall during the finals?” was asked thusly: “If student wrong venue then how?” The word “can” is used in the broadest sense possible, and always positive. “Is this item also on sale? Can!” (meaning yes). “Does anyone have extra cycles to spare as invigilators? Can!” (meaning I do). Sometimes “Also can” is used in the case of multiple cans and does not imply anything negative.
- Expect to be called “sir”, “mam”, and (very occasionally) “boss”. And of course “prof”. Don’t make the mistake of trying to get your students to not call you prof of their heads will explode.
- Getting in touch with born Singaporeans is very hard unless you are lucky to work with them. The crowd is very international and it is easy to get lost in the expat community.
- Expect to miss out on most things that happen around you as there is very little information available – you are simply supposed to “know” stuff. Read the notice boards. Talk to people. Network. But don’t ask students where to get good youtiao as they will become extremely anxious and worry that you weren’t pleased with their advice or couldn’t find the place even though you had it on Google Maps.

## 7 Going Forward

There are several things in the pipeline for going forward, which I briefly summarise below.

### 7.1 TUFF Grant 2020

I have received a 200k SEK grant for continuing the collaboration with NUS/SoC/Martin and in particular import some of the tools they use to streamline introductory programming education, in particular the Source Academy system.

### 7.2 SigCSE Grant for Mastery Learning

During the sabbatical, Martin and myself formed a working group on mastery learning in programming and software engineering education together with colleagues in Switzerland and Canada. We were able to get a small 5k USD seed grant for some initial activities. We hope to scale this endeavour up over time.

### 7.3 Future of SICP Book

The last week of the sabbatical, Martin and I finished and advertised the adaptation of the aforementioned classic course book used in CS1101s and several other places in the world. This adaptation is now being translated to Chinese, and will be pitched to MIT Press as a printed book.

The current state of the book is found here: <https://sicp.comp.nus.edu.sg/>

### 7.4 Head of Education

Since 1st of January 2020, I am the new head of education at the IT department at Uppsala University. This puts me in an excellent position to import my favourite ideas from NUS/SoC. I am hesitant to put to many of these in print, but I can mention a few things which I will be lobbying for:

- Setting up structures for making students part of developing software used in the education.
- Starting up a computing for welfare club.
- Increase transparency and link data sources together in relation to course evaluations and course results (NUS course evaluation reports were excellent, and e.g. compared CS1101s results with SoC average, NUS average, etc.)
- Introduce some equivalent of extra-modular credits to incentivise “advanced students”.
- NUS is great at displaying NUS pride, iterating core values, and celebrating successes. This is in dire need of importing.

### 7.5 Miscellaneous

- I hope to be able to visit Martin again in the not too distant future to help him with introduction of mastery learning in CS1101s.
- I hope to be able to host Martin in Uppsala for a sabbatical.
- I hope to make Uppsala a partner in the development of the Source Academy (the system running CS1101s).

## 8 Concluding Remarks

My biggest failure in my execution of the teaching sabbatical was to not do a good enough job of blocking time to think and mentally catch up. Do not repeat my mistake, or you will be sorry (too).

Nevertheless, it has been an amazing experience. Martin and Kok Lim made me feel like a valued peer from day 0. I bring home countless insights from NUS, CS1101s and working with Martin. So many teachable moments.