Writing Development Initiative
2021-2022 Proposal Application Form

Please answer all of the following questions as clearly and concisely as possible, filling in your responses immediately below each question.

When you have answered the questions, please indicate Chair approval in the space provided at the end of the form, and send the completed version of the form as a Word document to Michael Kaler (michael.kaler@utoronto.ca).

If you would like to see proposals from previous years, there is a selection here: https://www.utm.utoronto.ca/asc/wdi-archives.

If you have questions, please do not hesitate to contact Michael.

Deadline: Proposals must be submitted by **April 16th, 2021.**

1. Please indicate the course code:

   The Computer Science department would like to introduce small bits of writing instruction and assessments in the core first- and second-year CS courses. These writing instructions and assessments build on one another, so we would like to treat this project as a single WDI project. Here are the courses that are involved:
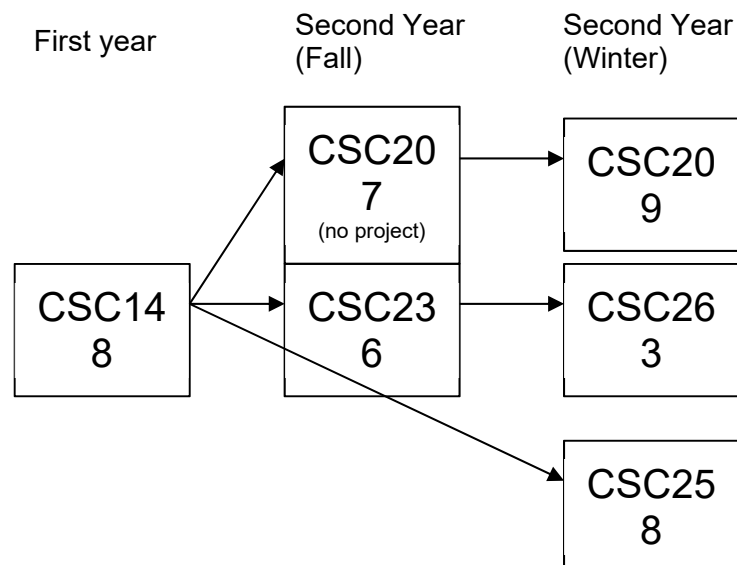
   CSC148 Introduction to Computer Science
   CSC236 Theory of Computation (prerequisites: CSC148 and MAT102)
   CSC209 Software Tools and Systems Programming (prerequisite: CSC148, via CSC207)
   CSC263 Data Structure and Analysis (prerequisite: CSC236)
   CSC258 Computer Organization (prerequisite: CSC148)

Figure 1: Core Computer Science Course Prerequisites

2. Please briefly (150 words maximum) introduce the course, its position in its program, and writing assignments or instruction that have typically been used.

The proposed courses provide coverage of the second year, allowing us to reinforce the writing instruction done in the first year and to extend the work to the various genres of writing that the students will encounter in the third and fourth years of the program. The quality of the writing is currently not assessed in any of these courses.

Note: In the past, CS students received writing instruction and feedback in CSC290 Communication Skills for Computer Scientists. This course is now discontinued, and first-year CS students will take ISP100 instead. This proposal replaces the previous WDI project in CSC290.

First-year Core Courses	Second-year Fall Term (Core Courses)	Second-year Winter Term (Students choose 1-3 courses)
CSC148 Students are currently required to write code comments (docstrings) as documentation that describes how their code works and explains their algorithms	**CSC207** Students work together in a group to build a small piece of software, and produce pieces of writing related project management (scrum logs, progress report). *Note: CSC207 is not included in this proposal since no changes are being proposed to the writing assessments and because the coordinator for the course has not yet been established.*	**CSC209** No writing is currently assessed.
	CSC236 Students write proofs, and explanations of their approach to solving a computer science problem.	**CSC263** Students write proofs, and explanations of their approach to solving a computer science problem.
		CSC258 No writing is currently assessed.

3. Please indicate the desired learning outcomes for the proposal (as distinct from the course as a whole), and how these learning outcomes relate to the course or program's learning outcomes: that is, indicate how the proposal complements student learning viewed holistically.

By the end of second year, we would like students to be able to have the communication skills required of a junior software intern or a junior research assistant, including:
- **Communicating effectively about intended and actual software behaviour (i.e., documentation and bug reporting)**
  - **CSC148/CSC209/CSC258**
- **Communicating the approach to solving a computational problem (i.e., commenting and proofs)**
  - **CSC148/CSC236/CSC263**
- **Structuring writing in effective (e.g., chronological, high-to-low level of detail, etc.) and expected formats (i.e., proofs, documentation)**
  - **CSC209/CSC236/CSC258/CSC263**
- **Use appropriate terms and tone for the audience (i.e., user vs. developer, level of specificity)**
  - **CSC209/CSC236/CSC258/CSC263**

4. Please provide a basic overview of the strategies that will be used to improve students' writing.

	Learning Objective	Instruction	Assessment
CSC 148	Differentiate between comments that describe the code (what it does), and comments that explain why the code is written in a particular way (how it solves the problem). Write clear comments and documentations explaining the function that a piece of code serves, and why it is necessary. Adhere to the expected format for comments and docstrings.	**Documentation** 60 minutes of tutorial instruction to discuss ● Different types of comments and their uses (summarizes a section of the code, explains why design choice…) ● Variable and function naming 30 minutes of **additional** instruction during lecture to debrief mid-way through the term.	The regular assignments (there are 2-3 of them in total) will have a documentation component. We will focus on A1 for formative feedback on writing skills. Writing TAs will assess the documentation quality in A1 and provide a grade as well as formative feedback. Writing TAs will assess writing in the A1 resubmissions to grade the quality of the student's improved documentation. The TAs will use the same rubric, and will provide writing feedback to students for these rubric items on the initial submission.
CSC 207	N/A		
CSC 209	*Note that these build upon what students learned in CSC148 about documentation.* Identify the structure of various sections of technical documentation and their role. Write technical documentation that is well structured and complete.	We currently use 20 minutes of lecture time and 30 minutes of practical time to introduce the format of linux documentation to support outcomes related to "reading documentation" and to encourage the use of documentation in self-learning. We can provide an additional 30 minutes of lecture time to expand	A technical documentation portion will be added to Assignment 1. This technical documentation will be due *before* the coding portion of Assignment 1, to introduce students to the idea of using writing to understand software specifications, and test-driven development. Writing TAs will assess the quality of technical documentation and provide

	Provide appropriate examples to illustrate the use of a piece of code.	on the required components of documentation and to build on instruction in 148 about picking appropriate examples for technical documentation.	writing feedback. Students will be required to re-submit the assignment with improvements based on the TA feedback. ● Structure/Organization ● Choice of Examples ● Tone/Clarity
CSC 236	*Note that this will build upon what students learn in the MAT102 projects, but specialized to computer science.* Write logically organized proofs and explanations. Use clear language. Clearly define and introduce new terms.	The instructor will prepare asynchronous videos about how to structure and write proofs in Computer Science. Lecture activities throughout the course will provide examples of: ● Correct vs Incorrect proofs ● Well vs poorly organized proofs ● Clearly written proofs vs proofs that are ambiguous ● Proofs with terms that are clearly defined, vs not.	Writing TAs will assess the writing quality of one random question in each of the three problem sets. The TAs will use the same writing rubric across all problem sets, and will provide writing feedback to students for these rubric items. The rubric and feedback will focus on ● Logic/Organization/Structure ● Clarity of language/terms
CSC 258	Use appropriate terms and figures to communicate with a non-expert audience. Avoid the use of jargon and assumed knowledge.	We will allocate 50 minutes of tutorial instruction on writing user guides or instructions. This tutorial will introduce the structure of non-technical user guides, appropriate use of visual aids, and methods for identifying and avoiding jargon and assumed knowledge.	A user-guide (documentation) requirement will be added to the open-ended end-of-term project. Writing TAs will assess the quality of the documentation and provide feedback.
CSC 263	*Note that these will build upon what students learned in CSC148 **and particularly 236.*** Write logically organized proofs and explanations, working at various levels of abstraction (e.g. code and math). Use clear language and appropriate tone. Clearly define and introduce new terms	50 minutes of tutorial instruction on structure and explanation of informal proofs, and explanations of ideas. This tutorial will ideally use similar rubric as the one CSC236, and include a peer review activity using excerpts that uses CSC263 content. The writing required is similar to CSC263, but is more advanced. Students are required to reason both mathematically and using code, and be able to communicate clearly while distinguishing between the two modes of thinking. Moreover, in CSC263 we will ask students to use appropriate tone to communicate professionally.	Students will be asked to complete a "take-home interview problem", which will be similar to an existing problem set question. Students will submit code solving a data structure design problem. More importantly, students will submit an explanation of why their solution works and has the desired runtime. This explanation is intended to be read by a potential employer or collaborator. Writing TAs will assess the writing quality, and the students will be required to re-submit the explanations with changes. The formative feedback will focus on ● Logic/Organization/Structure ● Clarity of language/terms ● Appropriateness of the tone

5. As of September 2020, UTM has begun offering a first-year writing course, ISP100H5 *Writing for University and Beyond: Writing About Writing*. For the 2021-2022 school year, this course will be required by the Departments of Anthropology, Chemical and Physical Sciences, Mathematics

and Computer Science, and Visual Studies for admission to some of their Specialist and Major programs. If you are proposing a project for a first-year course in any of these Departments, please be sure to consider how the project would complement or reinforce instruction offered in ISP100H5. For further details about ISP100H5, please contact Michael Kaler (michael.kaler@utoronto.ca).

The instruction provided by this WDI project builds upon the skills developed in ISP100, and complements ISP100 by providing Computer Science specific writing instruction. Moreover, we assume that students in second-year courses will have developed skills (like peer review) to assess sample student work during the tutorials.

6. Please indicate how Teaching Assistants will be used in the project.

We propose hiring and training a small number of "writing TAs" that will support the teaching and assessment of writing in multiple courses in the program. They will supplement the existing TAs assigned to the courses in the proposal.

	Tutorial Instruction	Benchmarking	Assessment
CSC 148	[1h/practical section] Lab instruction + [1hr/TA] Preparation	[2 hr/TA] benchmarking	[15min/student] Assignment 1: - 5 min to read the code + 5 min to formulate appropriate formative feedback. Assess the documentation quality in A1 and provide writing feedback to students. - 5 min to read the resubmitted documentation and provide a documentation quality grade. No feedback is provided on the A1 resubmissions.
CSC 207	N/A		
CSC 209	None: instruction will be performed in lecture time, using existing resources.	[2 hr/TA] benchmarking	[15min/student] Writing TAs will assess the adherence of the writing to the expected tone and structure of technical documentation. They will also assess the quality of the examples selected for inclusion. [5 min/student] Students will be required to re-submit the assignment with improvements based on the TA feedback. Writing TAs will verify that feedback has been incorporated.
CSC 236		[1 hr/TA] benchmarking	[20min/student] Writing TAs will assess the writing quality of one random question in each of the three problem sets. The students submit these either individually or in pairs. - PS1: 15 minutes to provide rubric feedback **and** personalized feedback for a problem in problem set 1 - PS2: 15 minutes to provide rubric feedback **and** personalized feedback for a problem in problem set 2 - PS3: 10 minutes to provide feedback (rubric only) for a problem in problem set 3
CSC 258	[1hr/tutorial section] Tutorial instruction + [1hr/TA] Preparation	[2 hr/TA] benchmarking	[20min/student] Writing TAs will assess the user-guide (walkthrough) of the additional features implemented in the open-ended end-of-term project. Feedback will be provided on the TA's experience using the walkthrough.

CSC 263	[1hr/tutorial section] Tutorial instruction + [1hr/TA] Preparation	[1 hr/TA] benchmarking	[15min/student] Writing TAs will assess the writing quality of the written explanations to the "take-home interview" problem [5min/student] Writing TAs will also assess the changes submitted by students.

7. Please indicate whether additional TA training (beyond the WDI Writing TA Training session for new TAs) will be required and, if so, indicate the number of hours/TA (maximum 4), content of the training, and its relationship to the proposed student assessment or instruction.

   We would like to have 2 hours of additional training per TA:
   - **2 hour of training on writing in computer science, and what "good feedback" looks like in CS courses**
   - **Benchmarking per assignment is included in #6 and #12**

8. Please describe the writing tasks incorporated as a direct result of the additional funding requested, and provide details on any writing instruction to be provided that relates to these tasks. If the funding is supporting an increased number of graded writing assignments, please indicate the number of additional words students will write.

First Year (Winter)	Second Year (Fall)	Second Year (Winter)
CSC148 - Docstring Additional ~300 words (Week 6)	**CSC236** Feedback on existing writing	**CSC209** Additional ~300 words (Week 5) Resubmission in Week 7 **CSC263** Additional ~100-200 words (Week7) Resubmission in Week 10 **CSC258** Additional ~1000 words (Week 12)

CSC148: Students document their code, describing both how it is used and how it implements the functionality internally. The WDI support would be useful for TAs to run a tutorial on writing good code documentation in written English. Additionally, writing TAs will provide feedback on the documentation which students submit with their assignment, in order to improve the students' written communication skills. The written documentation is estimated to be around 300 words, but may vary based on the assignment specifications.

CSC207: While this course is not formally part of the WDI proposal for this year, as no additional writing will be added and because the coordinator for the course has not yet been established, we anticipate that the "writing TAs" may be assigned additional hours from the TA resources already allocated to CSC207 to support the assessment of writing in a consistent manner.

CSC209: The course has a focus on interpreting and using technical documentation. Students will write documentation for a new library call in the appropriate format (a man page). This assignment is about 300 words.

CSC236: Students write proofs, and explanations of their approach to solving a computer science problem. No additional writing is assigned; the project provides feedback to existing writing.

CSC258: The course currently features an open-ended final project. A user guide describing the open-ended features implemented, fits naturally into the project. The guide would take the form of user documentation requiring about 1000 words. Since this is a final project, revision and resubmission is difficult, so a pre-submission workshop will be run in a lab to get students started and to provide formative feedback.

CSC263: Students will write an explanation of their approach to solving a data structure design problem. This type of explanation was required, but we anticipate that posing the explanation as a "take-home interview" problem will probe students to write longer and more clearly for such an audience, likely 250-500 words (additional 100-200 words).

9. Please clearly state the number of students participating in the project, if the proposed project is course-based. Indicate the maximum enrolment for the relevant course(s) and the final enrolment in the courses the last time they were offered. Please also indicate the course's relationship to the broader program of study.

   These are the course enrollments for this past year:
   https://student.utm.utoronto.ca/timetable/ March 16th 2021

	Course Enrollment
CSC148	861
CSC207	482
CSC209	305
CSC236	417
CSC258	356
CSC263	291

10. Please provide details on how the funded activities will impact and support students, if the proposed project is not restricted to a specific course (or courses).

    Once again, this project is intended to replace the existing WDI funded project in CSC290. We would like to spread the writing instruction out to multiple courses to highlight the importance of written communication in Computer Science. Moreover, the selected courses lead to two "streams" of writing with distinct ways that the project impacts students:

    CSC148-CSC209-CSC258: *Documentation* of code and process for both technical (CSC148, CSC209) and non-technical audiences (CSC258).
    CSC236-CSC263: *Proofs* and other documentation of logic or problem-solving approach for technical audiences.

With the current proposal, students will receive 15 minutes of writing instruction in the first year, and an average of approximately ~50 minutes of writing instruction in the second year.

The latter number is estimated using the past year's enrollment numbers. (Note that we have had some disruptions this year due to COVID, so the distribution could be different next year)

	Count	Writing Support
#Students who took CSC236 only	**124**	**20min**
#Students who took CSC236 + One of CSC209/263/258	**130**	**40min**
#Students who took CSC236 + Two of CSC209/263/258	**82**	**60min**
#Students who took CSC236 + Three of CSC209/263/258	**107**	**80min**

11. Please indicate any other resources you will use to support your project (library, RGASC, online resources, etc.).

   We would like the RGASC to run drop-in times for second-year students in the winter term, to support the following:
   - **CSC209, to support the resubmission of the writing assessment (Around Week 7)**
   - **CSC263, to support the resubmission of the writing assessment (Around Week 10)**
   - **CSC258, to support the writing assessment (Around Week 12)**

   We would also like assistance/feedback on instructional materials and assessment rubric.

   We would like to hire writing TAs who will work in multiple of our courses, and would like help attracting graduate TAs who may not necessarily be studying Computer Science.

   We would also be interested in a WDI Assessment of how student writing evolves across these courses, to help evolve the writing instruction in these courses.

12. Please provide a detailed budget.

	Instruction Hours	Assessment Hours	Total Requested Hours
CSC148	1 hour / TA for prep for the writing tutorial * 6 TAs = 6 hours 1 hour / TA for writing tutorial * 8 timeslots per TA * 6 TAs = 24 hours Total: 30 hours	15min/student * 900 students = 225 hours Benchmarking (consistency check-in): 2h per TA * 6 TAs = 12 hours Total: 237 hours	267 hours

CSC207	N/A		
CSC209	None	20 min/student * 300 students = 100 hours Benchmarking: 1 hr/TA *6 TAs = 6 hours	106 hours
CSC236	Instructor-lead instructions only	20min/student * 400 students = 134 hours Benchmarking (consistency check-in): 2h per TA * 2 TAs = 4 hours Total: 138 hours	138 hours
CSC258	1 hr/TA for prep * 2 TAs = 2 hours 1 hr/tutorial *10 tutorials = 10 hours	20 min/student * 360 students = 120 hours Benchmarking: 1 hr/TA * 6 TAs = 6 hours	124 hours
CSC263	1 hour / TA for prep for the writing tutorial * 3 TAs = 6 hours	20min/student * 300 students = 100 hours Benchmarking (consistency check-in): 1h per TA * 6 TAs = 6 hours Total: 106 hours	106 hours
TOTAL			**742 hours**

13. Please include this sentence in your application: "I confirm that I approve this proposal."

I confirm that I approve this proposal.

These are the instructors who we expect will be coordinating the relevant courses:
- **CSC148: Bogdan Simion**
- **CSC236: Michael Miljanovic**
- **CSC209: Andreas Bergen**
- **CSC258: Andrew Petersen**
- **CSC263: Lisa Zhang**

I confirm that all instructors involved in these courses approve this proposal.

Lisa Zhang and Bogdan Simion will serve as the point-persons for the project as a whole and will be responsible for liaising with the WDI Committee. We will ensure that instructors are on board and we will be gathering reports from the instructors to aggregate into a Final Report for the committee.

14. Please also include this sentence in your application: "I confirm that my Chair supports this proposal."

I confirm that my Chair supports this proposal.

(Our Associate Chair also supports this proposal.)