UNIVERSITY OF CALGARY | FACULTY OF SCIENCE

CURRICULUM REVIEW REPORT

DEPARTMENT OF COMPUTER SCIENCE

Table of Contents

Curriculum Review Team

Review Leads: Benjamin Stephenson, Nathaly Verwaal
Faculty: Sonny Chan, Charles Hepler (MRU), Leonard Manzara, Rob Walker, Wesley Willet
Graduate Students: Zain Rizvi, Masoumeh Shafieinejad
Undergraduate Students: Ronald Chang, Emily Chow
Staff: Darcy Grant, Susan Lucas

Executive Summary

The following undergraduate programs have been assessed in the curriculum review: BSc in Computer Science, BSc Internship in Computer Science, BSc Honours in Computer Science, and BSc Internship Honours in Computer Science programs.

Curriculum Review Process

The review mapped course learning outcomes to program learning outcomes and to Faculty-level graduate attributes to determine whether the delivery of the program matches these outcomes and attributes. It also asked four guiding questions to identify strengths in the program and find areas where improvement is possible. The curriculum review was guided by two program specific guiding questions:

1. What are the current barriers to effective teaching and learning in our undergraduate program?
2. How should our department (faculty, staff, space, resources, etc.) and programs change over the next decade so that we can continue to offer a relevant, high quality and valuable degree that will attract a strong and diverse set of students to this institution?

In addition, two guiding questions are being asked by the Faculty as a whole.

3. Are High Impact Practices being used regularly in this program?
4. If not, what is preventing these practices from being used?

Strengths of the Computer Science Programs

The Computer Science programs are high-demand programs that attract highly capable students (minimum high school average required is 84%). Retention rate in the programs are currently very high (95–100%) and students find positions in local, national, and international companies ranging from oil and gas, entertainment industries, tech companies, and software development companies.

The Department of Computer Science is a research intensive department. Students are integrated into the research initiatives through a variety of courses and research opportunities. Students are exposed to recent research and developments in industry through the wide variety of 4th year courses offered, especially special topic courses (CPSC 599).

Students are encouraged to enhance their Computer Science programs through interdisciplinary studies especially in some concentrations and through the Computer Science minor.

The Department (and its programs) promote international experiences through its participation in a Faculty-level exchange program and the 2+2 programs.

Concerns in the Computer Science Programs

The program learning outcomes and program delivery needs to align closer with student careers. Students report that some of the skills they learned are not needed for their careers yet some skills that would have been helpful are missing. An adjustment is needed to provide a more flexible program that can be catered to both research-intensive and practice-intensive paths in a variety of areas. Furthermore, to give students practice to integrate what they have learned the goal is to see if we can design a capstone or research experience that we can offer all our students rather than the small number of students we accommodate currently in capstone and research courses.

There has been a marked increase in student enrollment in the last five years which has led to unpredictable class sizes. This need to be managed in a more consistent, predictable manner.

Student engagement with the curriculum review was exceedingly low and this echoes faculty concerns about lack of student engagement in coursework and outside of courses.

Goals for the Computer Science Programs

Based on the review's findings, the Department will address the following key goals over the next 5 years.

1. Align the Computer Science programs closer with student careers.
2. Design a capstone or research experience for all students in the Computer Science major programs.
3. Improve student engagement.
4. Manage increase in student demand for the Computer Science programs and courses in a more consistent, predictable manner.
5. Improve the curriculum review process.

Overview and Context of the Program

The Department of Computer Science offers two core undergraduate programs: a four-year Bachelor of Science in Computer Science and a four-year Bachelor of Arts in Computer Science, the second of which is a combined degree with the Faculty of Arts, and is currently undergoing review. The Department also offers BSc Honours, BSc Internship, and BSc Internship Honours programs, plus a combined BComm/BSc program in conjunction with the Haskayne School of Business. This document focuses on the BSc, BSc Internship, BSc Honours and BSc Internship Honours program.

Our BSc programs resemble what is now viewed as a traditional program in Computer Science, with a core of required courses in Computer Science, Mathematics and Statistics, Software Engineering, and Philosophy. Undergraduate students have the option to complete a small number of additional courses, in areas of strength for the Department, in order to complete a concentration that is noted on their transcript. Concentrations that are presently available include:

- Computer Game Development
- Software Engineering
- Theoretical Computer Science
- Scientific Computation
- Human-Computer Interactions
- Computer Graphics
- Information Security
- Networks and Distributed Computing

Historical Context

In its early days at the University of Calgary, Computer Science was part of the Department of Mathematics, Statistics, and Computing Science, which was administratively within the Faculty of Arts and Science. A logical split of this department occurred in 1975, creating the Department of Computer Science separate from the Department of Mathematics and Statistics. At the time of creation, the Department had 14 full-time academic staff, 2 support staff, and a good balance between theoretical and applied aspects of computing.

Significant growth and change have occurred over the past 4 decades. Some of the highlights include:

- launching an Internship Program in 1990 to broaden the academic mandate of the Department and respond to the needs of local industry;
- doubling the size of our academic staff and our student enrollment in the late 1990's as part of a major provincial expansion in ICT;
- substantial increases in graduate student numbers, research funding, and international research impact as a result of our expansion; and
- relocating (most of) the Department to the new Information and Communication Technologies (ICT) building in 2001.

Program Learning Outcomes

The Department of Computer Science has formally adopted the Canadian Information Processing Society's recommended competencies for graduating students:

1. **Demonstrate Knowledge**. Competently apply knowledge in
   a. *software engineering*: including requirements specification, software design and software architecture, software development, software testing, software maintenance, and other topics related to software process;
   b. *algorithms and data structures*: including data structures such as stacks, trees, lists, queues, etc,; abstract data types, established solutions to classical problems (e.g., sorting and searching), and analysis of algorithms;
   c. *systems software*: including operating systems concepts, virtual memory management, management of distributed, parallel, and concurrent processes, transaction processing, logging, security, and computer networking;
   d. *computer elements and architectures*: including computer organization, digital device and communications technology, logical and physical hardware design;
   e. *theoretical foundations of computing*: including models of computation, analysis of algorithms, fundamentals of program specification and verification, computational complexity, grammars and automata;
   f. *discrete mathematics*; and
   g. *probability and statistics*.
2. **Analyze and Solve Problems**. Use appropriate knowledge and skills, including background research and experimentation, to identify, investigate, abstract, conceptualize, analyze, and solve complex computing problems, in order to reach substantiated conclusions.
3. **Design Software and Systems**. Design and evaluate solutions for complex open-ended computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, as well as economic, cultural, societal, and environmental considerations
4. **Use Appropriate Resources**. Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of their strengths and limitations.
5. **Work Individually and in a Team**. Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings.
6. **Communicate Effectively**. Communicate with the computing community and with society at large about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions
7. **Act Professionally**. Act appropriately with respect to ethical, societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and with regard to the consequential responsibilities relevant to professional computing practice.
8. **Be Prepared for Life-Long Learning**. Learn new tools, computer languages, technologies, techniques, standards and practices, as well as be able to identify and address their own educational needs in a changing world in ways sufficient to maintain their competence and to allow them to contribute to the advancement of knowledge.
9. **Demonstrate Breadth of Knowledge**. Possess knowledge in areas other than computer science and mathematics so as to be able to communicate effectively with professionals in those fields.

Guiding Questions

Program-Specific Questions

The curriculum review was guided by two questions that are specific to the Computer Science program:

1. What are the current barriers to effective teaching and learning in our undergraduate program?
2. How should our department (faculty, staff, space, resources, etc.) and programs change over the next decade so that we can continue to offer a relevant, high quality and valuable degree that will attract a strong and diverse set of students to this institution?

Faculty-Wide Questions

In addition, two questions are being asked by the Faculty as a whole.

3. Are High Impact Practices being used regularly in this program?
4. If not, what is preventing these practices from being used?

Action Plan

Based on the review's findings, the Department will address the following key goals over the next 5 years.

1. Align the Computer Science programs closer with student careers.
2. Design a capstone or research experience for all students in the Computer Science major programs.
3. Improve student engagement.
4. Manage increase in student demand for the Computer Science programs and courses in a more consistent, predictable manner.
5. Improve the curriculum review process.

The action plan to achieve these goals is as follows.

Recommendation	**Action Item**	**Who is Responsible?**	**Due Date**
Ensure program learning goals match skills needed by students in the various careers they pursue after graduation.	• Create measurement tool to assess effectiveness of program and course changes to meet goal. • Draft learning outcomes for each concentration in computer science seeking input from department members, students and industry advisors. (Meant to guide identification possible 'streams' of study in Computer Science.) • Update program learning outcomes to ensure appropriate balance for preparation for research-intensive and practice-intensive career paths. • Verify if learning outcomes still align with outcomes for CIPS accreditation. • Update required courses for Computer Science programs and concentrations based on updated program and concentration learning outcomes. • Note: these change will probably start with first year courses in 2017, then second year courses in 2018, etc.	Associate Head Undergraduate Affairs, Undergraduate Affairs Committee	September 2020
Improve access and exposure for all students	• Attempt to develop a sustainable plan to provide all students with a research or capstone experience in their final year of the program.	Associate Head Operations, Associate Head Undergraduate Affairs,	September 2018

to research opportunities and high impact teaching practices.	• If feasible plan is found, update all computer science major programs to make this a required part of the program.	Undergraduate Affairs Committee	
Manage increase in student enrollment.	• Develop a sustainable plan to distribute current teaching resources between required CPSC courses, optional CPSC courses and service CPSC courses.	Associate Head Operations, Associate Head Undergraduate Affairs, Undergraduate Affairs Committee, Associate Dean	September 2017
Improve student engagement.	• Create measurement tool(s) to assess student engagement. • Consult with students to find out how students wish to engage with the department, faculty and university.	Associate Head Undergraduate Affairs, Outreach Committee, Computer Science Undergraduate Student (CSUS) club	Ongoing
Improve curriculum review data collection.	• Maintain current course learning outcomes and course mapping information and update with program and course changes. • Engage with Faculty of Science to streamline data collection in next curriculum review.	Associate Head Undergraduate Affairs, Undergraduate Affairs Committee, Course Instructors	Ongoing

Conclusion

The curriculum review has shown that programs offered by the Department of Computer Science are high demand programs that offer high value to students. Students and faculty members take pride in the work they do inside and outside of class.

The Department aims to improve the undergraduate programs offered to align learning closer with student careers and to improve student engagement with the department and programs. Specific goals include:

1. Align the Computer Science programs closer with student careers.
2. Design a capstone or research experience for all students in the Computer Science major programs.
3. Improve student engagement.
4. Manage increase in student demand for the Computer Science programs and courses in a more consistent, predictable manner.
5. Improve the curriculum review process.

While the curriculum review has identified specific challenges to overcome on the route ahead, one must take the results with a grain of salt. Response rates were consistently low to the surveys, and respondents at the student and alumni levels have a necessarily limited perspective. However, it is clear that improvement is needed in communicating course- and program-level learning outcomes to all stakeholders, in explaining the rationales behind those learning outcomes, and in achieving those learning outcomes effectively.