McGill | School of Computer Science

# Celebrating 50 years on top

July 2022 Issue

# In this Issue



# Message from the Director

2021 marked the 50th anniversary of *your School,* McGill's School of Computer Science. The achievements of the School, its students, professors, and staff, are worth celebrating! While COVID threw a wrench into our celebration plans, the sky is clearing again, for what we will geekily call the $(50+e_{COVID})^{th}$ anniversary. This newsletter gives a good idea of where the School is at today, but it is worth reflecting on the path that took us here.

Initially launched by a number of visionary—if slightly rebellious—scientists, the School quickly became famous for the quality of the training it provided to its select students and the impact of its research. Initially centered on a core group of rowdy but world-leading theoretical computer scientists and computer systems researchers (remember Archie, the world's first internet search engine?), the School embraced and pioneered new research directions enabled by its strong foundations, including artificial intelligence and machine learning, software engineering, robotics, and biomedical applications.

Through the highs and lows of Computer Science (recall the dot com bust?), the School has maintained its drive for excellence and successfully captured opportunities to push the limits of the field while broadening its impact on society. Since its creation, more than XX students have received an undergraduate degree from the School, XX Masters, and XX PhD. Imagine the impact this has had on the development of the city of Montreal, the province of Québec, and Canada. Today, the School stands large and strong, ranking, for the first time and despite our moderate size, among the top 50 Computer Science departments in the world. Equally notable is the fact that the School trains the most diverse group of students in Canada, with women forming 37% of the student body.

We are thrilled to offer you this Newsletter to enable you to catch up with your School, and hope to see many of you at some of the many celebration events planned in 2022-2023.

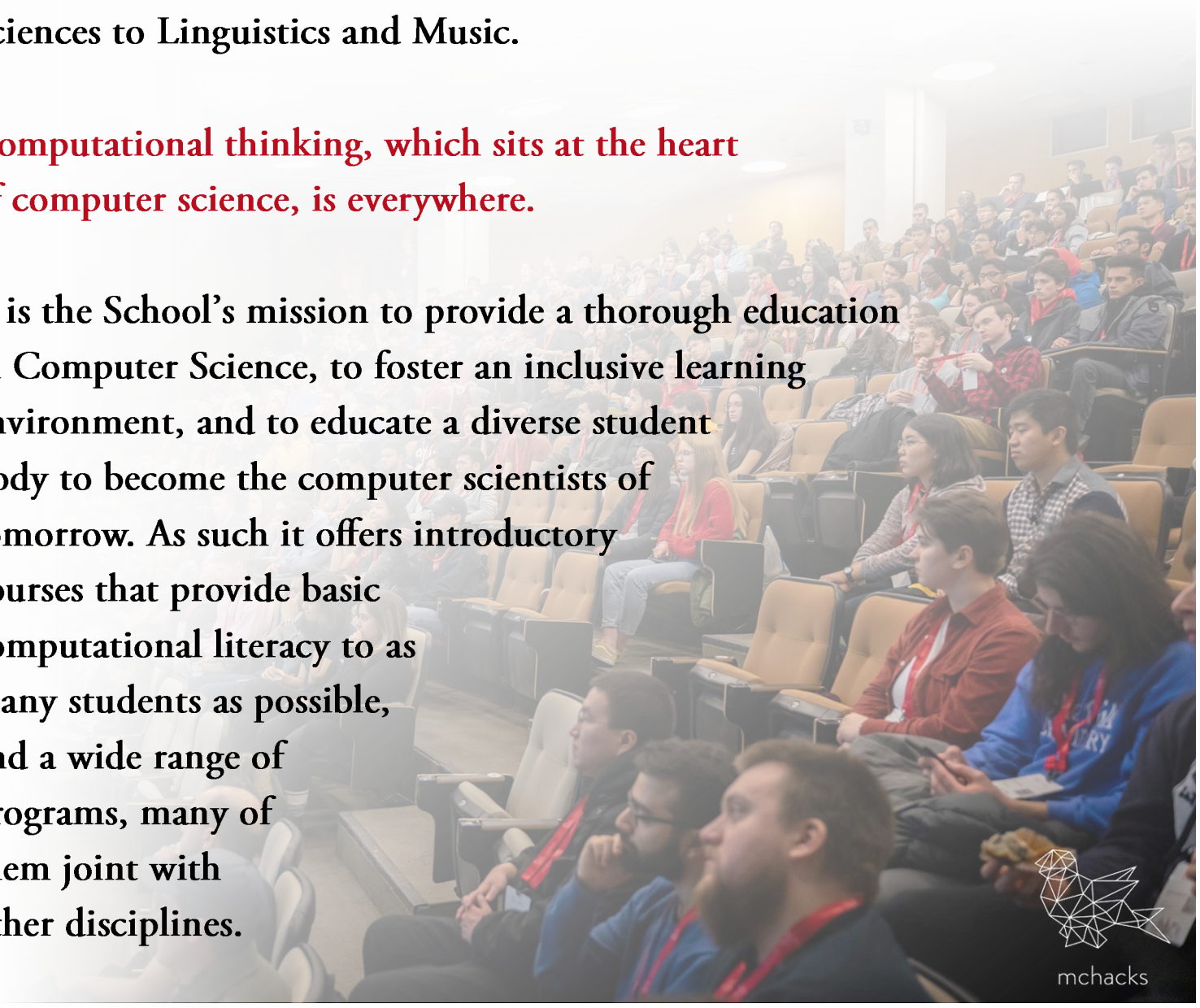
Mathieu Blanchette
Director, School of Computer Science

# Update on our Undergraduates

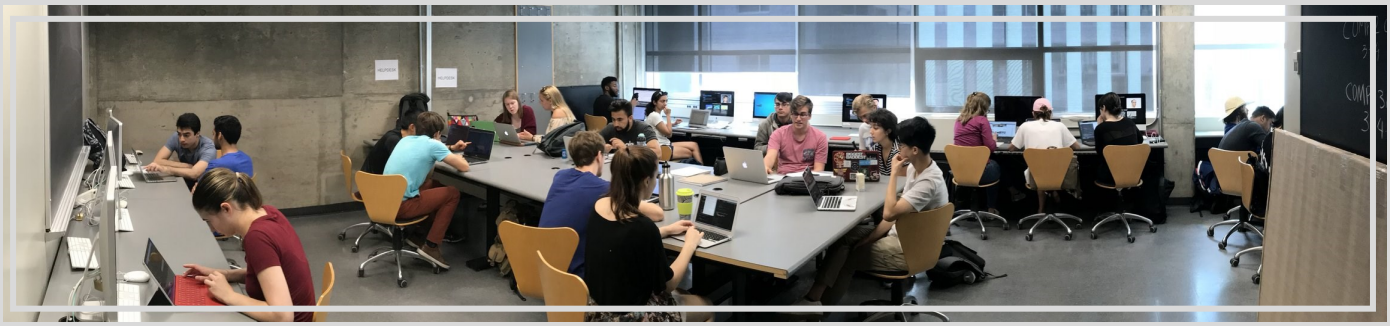
Computer science plays a pivotal role in our economy & culture today. It impacts and shapes many areas from Medicine and the Life Sciences to Linguistics and Music.

**Computational thinking, which sits at the heart of computer science, is everywhere.**

It is the School's mission to provide a thorough education in Computer Science, to foster an inclusive learning environment, and to educate a diverse student body to become the computer scientists of tomorrow. As such it offers introductory courses that provide basic computational literacy to as many students as possible, and a wide range of programs, many of them joint with other disciplines.



# Teaching Computer Science to Everyone



More than 10 years ago, the School recognized the vast impact computational thinking plays not only in Science and Engineering but also in Arts, and restructured the undergraduate program to provide opportunities for student in Science and Arts to study computer Science. This led to a highly interdisciplinary program structure that draws students from a variety of backgrounds. The success of this program design is clearly visible today in our enrolment.

Students not only recognize the impact of computer science itself, but also realize the impact of computational thinking and ideas on their own field of study. Over the past decade, we have seen the total number of student course registrations per year more than double from 5,500 in 2013-2014 to 13,225 in 2020-2021. Interestingly, our introductory programming courses (COMP 202/204/208), that teach basic programming and computational skills, have seen a less extreme increase in enrolment from 1,550 to 2,145. This means the increase is mainly due to the follow-up courses, indicating that students now tend to take more computer science courses. Presently, out of roughly 7,500 students enrolled in the Faculty of Arts, over 900 students are enrolled in a Computer Science program (major concentration/minor concentration/cognitive science), meaning more than 12% of our Arts students are taking Computer Science programs. Out of nearly 4,500 students enrolled in the Faculty of Science, roughly 1,650 students are either doing a major or minor in Computer Science. As a consequence, 37% of students in the Faculty of Science are enrolled in Computer Science programs. This shows the enormous impact our teaching computer science principles has on our overall student body. Our mission is that within 10-15 years, every student graduating with a B.Sc. degree knows the basics of how to write computer programs and understands the main principles of computer science.

# Commitment to Diversity & Equity

We have a longstanding policy of allowing any McGill student into any of our courses, provided the student has the course prerequisites. The main reason for this policy is that we do not want to create a barrier for underrepresented groups who might not initially see themselves as fitting in.

We are proud to have 37% of women in our major in Computer Science programs in the Faculty of Science! This is one of the highest numbers of women majoring in Computer Science programs in Canada, and stands in stark contrast to the North-American average of women in CS programs where only 21% of Computer Science degrees are awarded to women.

# Undergraduate Computer Science Research

Undergraduate Research plays an important role in the School of Computer Science. The capstone is the School's Undergraduate Computer Science Research Conference (UCORE) each summer. It offers undergraduate students who have been involved in Computer Science summer research at McGill the opportunity to submit and present their research. This allows students to learn about the breadth of computer science research and share their accomplishments with a wider audience. Many of these students continue their education in top graduate programs and some have become professors themselves.



Over the last three years, the School of Computer Science has witnessed an increase in the number of our students involved in competitive programming teams. Through this participation, students from different backgrounds and coding abilities have earned algorithmic experience as well as confidence in both their computing and interpersonal skills. Furthermore, this collaborative environment has fostered a welcoming and inclusive culture in which students have found a safe space to train and to share their competitive hobby with friends and classmates alike. Competitive programming activities have proven especially vital during the COVID pandemic. Having consistent work to do, meetings to attend, and contests to compete in has served to break the cycle of isolation amongst students.

McGill's programming contests are attracting hundreds of students and competitors, and McGill contestants have gone on to represent McGill in various international competitions. Among other honors, McGill qualified for the International Collegiate Programming Contests (ICPC) world finals during 2020 and 2021. ICPC boasts over 50,000 student participants representing some 3,000 universities in 111 different countries, all competing for just 100 final spots. Two years in a row, McGill has assembled a qualifying team. Next year, we aim to welcome even more talented students to take part in our competitive programming– perhaps you will represent us in the next world finals!

# Undergraduate Students

The **Computer Science Undergraduate Society** is a very active student group. Pre-COVID is organized major social events such as the highly anticipated Wine & Cheese at the beginning of each semester. CSUS also maintains an academic guide written by students which is complementary to the faculty advising. They also run an Industry Mentorship Program which aims to connect McGill computer science students with a mentor in industry.

**The CSUS Help Desk** is a team of over 40 undergraduates who offer tutorial and help for any of the COMP 2xx courses five days a week. This supplements the office hours given by Teaching Assistants. In fact, the Help Desk has developed their own web-based application that allows scheduling appointments. This initiative is co-sponsored through the School and the Tomlinson Engagement Award for Mentoring (TEAM) from McGill.

**McWics** (McGill Women in Computer Science) is a student-run group whose goal is to cultivate community for collaboration, exploration, and expanding opportunities for women in computer science. They organize site-visits to companies, run mock-interviews, organize hackathons for women students and organize informal meetings where women undergraduate, grad ate and postdoctoral students mingle with women professors.

## Faculty Lecturers @ SOCS:

The School is blessed with a team of full-time faculty lecturers who have decided to make teaching their career. They excel at teaching many of our largest undergraduate courses while bringing innovative approaches to course delivery taking advantage of the latest technologies, truly raising the bar in terms of standards. Behind the scene, our faculty lecturers also play essential roles in the School's everyday operations – nothing would work without their help!

**Joseph Vybihal** (Masters in AI and McGill) is our most experienced faculty lecturer. Since joining the School in 2000, he has taught nearly 5,000 student in areas such as introductory programming, computer systems, and web development, among others. He was also the leader of the highly successful CS Summer Camp series that ran for 13 years and attracted many bright young minds to CS and McGill!

**Giulia Alberini** (McGill PhD in cryptography), joined us four years ago and has since revolutionized how we teach our introductory courses. The way she teaches those intro courses, making them fun and challenging rather than dry and technical, is in large part why so many McGill students – in particular women - are migrating toward a degree in computer science. She's adored by her students who find her energy and passion communicative. She recently received the Principal's Prize for Teaching.

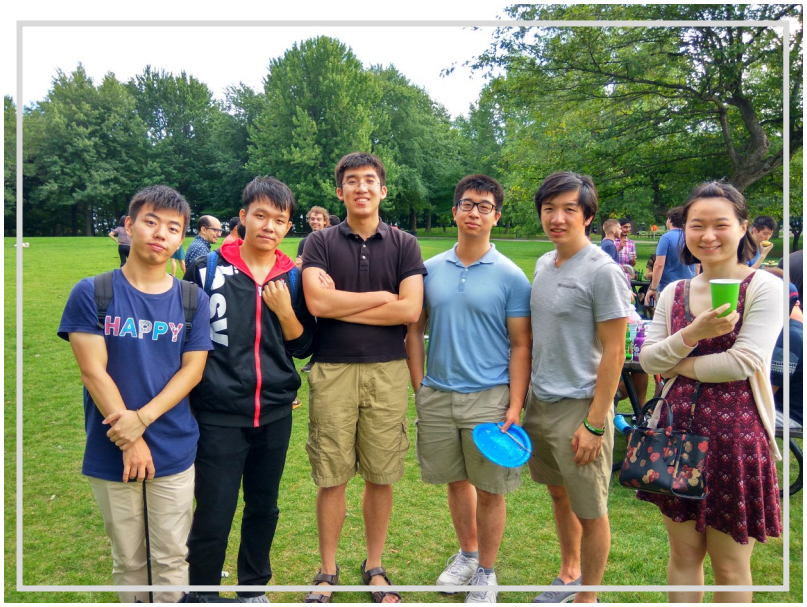
**David Becerra** (McGill PhD in bioinformatics), has a visible passion for teaching. Constantly on the lookout for ways to improve our courses, he is often the first to test and adopt new teaching technologies. Since 2017, he is also the coach of our highly successful teams of Competitive Programming, and has brought several of our teams all the way to the ICPC world's finals (a.k.a. the Olympic games of programming).

**Joseph D'Silva (**McGill PhD in distributed systems), is our expert in the field of computer systems (databases, operating and distributed systems). He has been effectively teaching courses with a high degree of technical content and managing to make the courses exciting and attractive. At a time where everybody and their uncle wants to learn AI, he has succeeded at making other branches of CS equally exciting to our students.

Our two latest recruits join the School in 2022. **Faten M'hiri** (PhD in medical imaging at ETS) and **Jacob Errington** (Masters in programming languages at McGill) are bringing fresh ideas and a ton of energy to our teaching team. We are lucky and thrilled to have them join our team.

# Graduate Affairs

Graduate students at both the PhD and the MSc level and postdocs are an integral part of the School's research and teaching mission. The School aims to provide its graduate students a well rounded program with strong breadth and depth components. We offer a wide range of graduate courses, and graduate students and post-docs work closely with their supervisors, peers and undergraduate students in research labs to generate the new knowledge that will drive the computer science of tomorrow.




Our graduate program is boosted by a highly active research program that is supported by many industrial partners as well as federal and provincial funding agencies. Many professors from the school hold affiliate positions in the industrial research labs and research institutes in Montreal. Through these partnerships the professors can find support for graduate students to carry out their graduate research.

We take pride in having an MSc program through which students are exposed to high quality research work in all areas of computer science ranging from theoretical topics to practical problems. This prepares the MSc students to take advanced positions in industry or move on to pursue PhD programs at top universities throughout the world. Our PhD students are trained to do cutting-edge research that results in publications in top venues. At the end of their degrees, the PhD graduates are capable of forging their own research agendas and transitioning to the next stage of their careers be that a post-doctoral, faculty, or industry position.

## Grad Students by Numbers

The School's graduate programs boast over 275 MSc and PhD students as well as approximately 15 post-docs. The number of PhD students has increased significantly over the last 5-10 years. This increase is due to the School being home to numerous large research groups in various areas of computer science including but not limited to bioinformatics, robotics, machine learning, artificial intelligence, computational theory, and distributed systems. Nearly 65% of our students are internationals, bringing a diversity of experience and thought to our graduate programs.

Our programs at both the MSc and PhD level remain highly sought after with over 2000 applications submitted each year over the last several admission cycles. Less than 10% of applicants are admitted due to capacity limitations. To the PhD program we primarily admit students with an MSc in Computer Science or equivalent. Recently, we have encouraged direct admissions from BSc programs for students with strong research experience in the relevant research areas. For the MSc program we have a BSc/MSc fast track option that enables undergraduates with strong research experience to start their MSc work sooner.

Most of our graduate students are enrolled in the program as full-time students. Some students begin their degree as full-time students and complete it while maintaining a part-time status, which allows them to commence a full-time job prior graduation.

Our PhD students receive competitive funding in the form of research scholarships and teaching assistantships. In addition, several of our students receive highly selective named fellowships that are sponsored by external agencies or corporations. Over the last several years, numerous industrial players in computer science have started labs in Montreal which have strengthened an already thriving gaming and information technology industry. This is making it easier for graduate students to find internships, some of which are based on the graduate research of the students and partially funded by research agencies.

# In Memoriam

Over the past few years, we have mourned the loss of valuable members of our School of Computer Science family.

Here, we honour and remember their contributions to our community, spotlight their lives and legacies, and give our sincerest condolences to their families and loved ones.

1988-2020

# Faraz Falsafi

Born in Tehran, Faraz Falsafi lived in Montreal for several years, earning his master's degree in Computer Science in 2014, and had since moved to Toronto to pursue his career. Falsafi had traveled to Iran to attend his sister's wedding and was victim to the January 2020 plane crash.

He is remembered as a skilled computer engineer who loved exploring forests like Mont-Tremblant provincial park, and spending hours outdoors in search of the perfect photo. Falsafi's immediate supervisor during his time at a Montreal start-up, our own Professor Joseph Vybihal, describes Faraz as having been an intelligent, hardworking, and resourceful man.

Falsafi's gentle, kind spirit is remembered fondly by the Montreal Computer Science community, and we extend our condolences to his loved ones.

1958-2019

# Laurie Jane Hendren

Professor Laurie Hendren passed away on the 27th of May 2019 after a long struggle with cancer. She is survived by her daughter Jane Panangaden, her brother Paul Hendren, and her husband Prakash Panangaden who is also a professor in the School of Computer Science. She was a brilliant researcher, teacher, and role model for women in computer science. Apart from her academic achievements, she was a fun-loving person and a good friend to many people not least her students. Her loud infectious laugh will never be forgotten.

Professor Laurie Hendren was an enthusiastic ice hockey player, a keen amateur musician, and a tough negotiator. Born in Peterborough, Ontario, Professor Hendren earned her undergraduate degree at Queen's University and her PhD from Cornell University in the US in 1990.

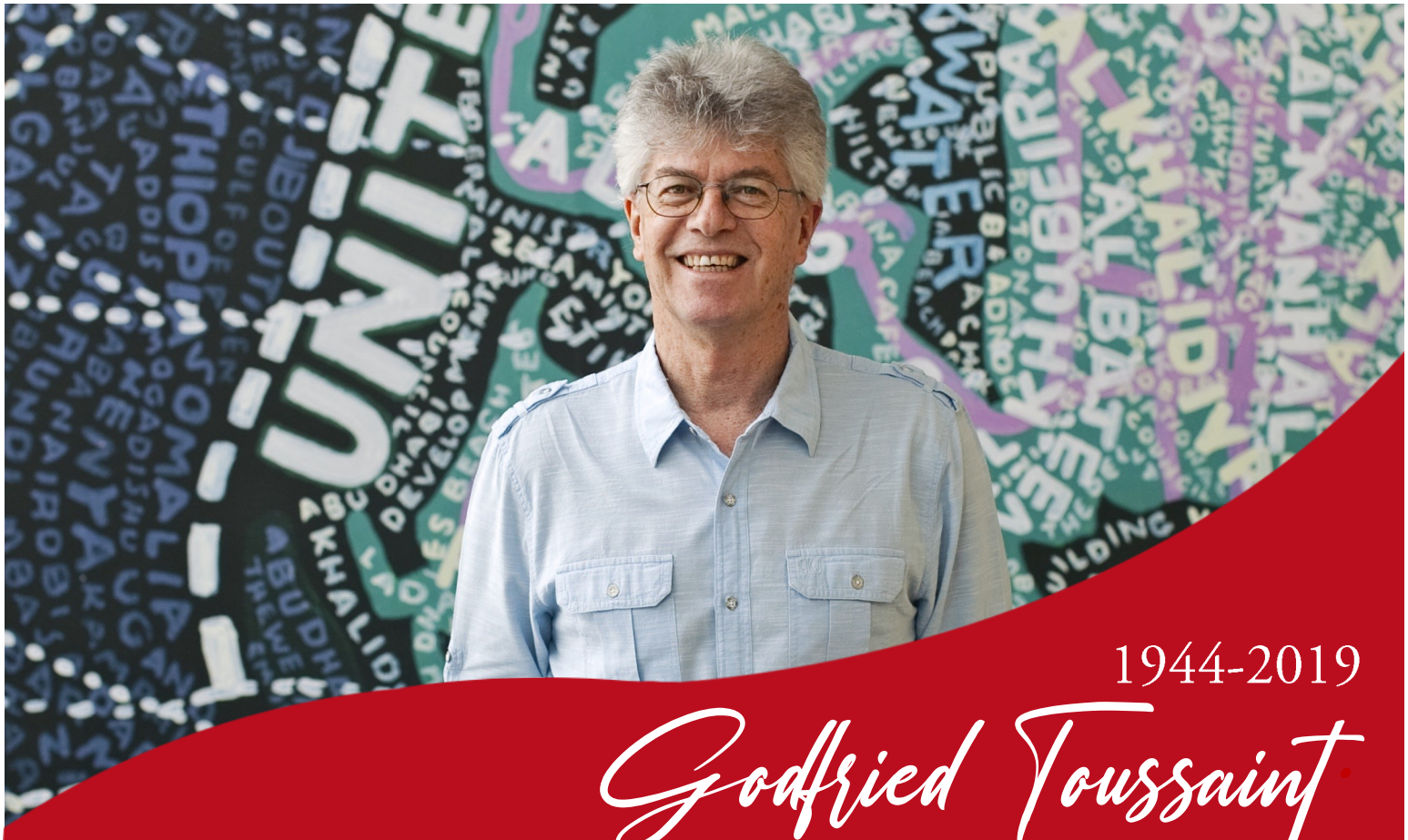
In her thesis, she developed new program analysis methods for detecting aliasing and interference in programs manipulating pointer-based dynamic data structures. Thereafter she worked on program analysis, parallelizing compilers, aspect-oriented programming, and applications of program analysis to software engineering. She received numerous accolades for her research, culminating in

the award of the Dahl-Nygaard Prize which was awarded shortly before she died. She was a Full Professor of Computer Science and held a Canada Research Chair in Compiler Tools. She was elected a Fellow of the ACM in 2009. She was elected to the Academy of Science of the Royal Society of Canada in 2012. She was posthumously awarded the Lifetime Achievement Award of the Canadian CS Research Community (CSCan-InfoCan).

She was Associate Dean (Academic) in the Faculty of Science and made substantial changes to the curriculum, which have had a lasting impact on undergraduate education. She was an outstanding teacher and was awarded the Leo Yaffe Prize for outstanding teaching in the Faculty of Science in 2006.

Professor Hendren leaves a lasting legacy in the form of her Opal project with the McGill University Health Centre. Informed by her own experience with cancer care, together with her former oncologist, a medical physicist at the MUHC, and a team of McGill students, Professor Hendren began to develop the Opal app to aid patients with long wait times. The project expanded to provide blood-test results, check-ins, and other vital information to patients. Although still in its infancy, Opal has already won a number of awards, including a Prix de cancerologie, the 2019 Prix d'excellence du Ministere de la Sante et des Services Sociaux, and several other prizes. Today, over 600 MUHC cancer patients are serviced by the app, with plans to expand to many other clinics and hospitals.

In her nearly 30 years as a member of the School, she created a lasting impact. Her husband has endowed a graduation prize and a scholarship in her name to be awarded for the first time in 2021 to coincide with the 50th anniversary of the founding of the School of Computer Science.




1944-2019

# Godfried Toussaint

Professor Godfried Toussaint passed away in July 2019 at the age of 75. He collapsed while giving a talk at the International Cartographic Conference in Tokyo, Japan, and passed away soon after. He is survived by his beloved wife Eva, and their daughter Stephanie.

Professor Toussaint was an outstanding researcher and teacher, widely considered to be the Canadian father of computational geometry. He is most known for his truly pioneering work in mathematical research in music, in particular music rhythm, but also for significant contributions in information theory, music information retrieval, and pattern recognition and analysis.

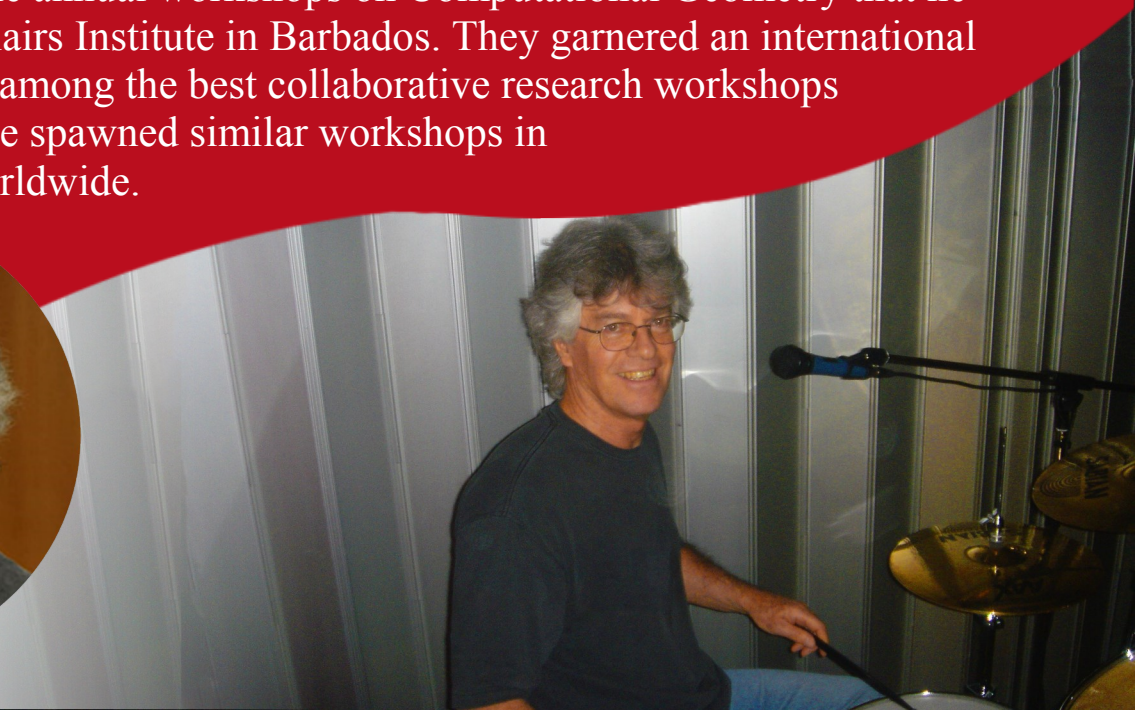
Toussaint was born in Belgium and completed his undergraduate studies at the University of Tulsa in 1968, followed by his PhD at the University of British Colombia in 1972. He joined the School of Computer Science in 1972, just one year after its creation. In 2005, he became a Collaborator with the Schulich School of Music, and in 2007, he became Emeritus Professor. Over the decades, he was one of the pillars of the School and an immensely influential member, significantly contributing to our becoming one of the leading computer science research

departments in Canada. He was good-natured, helpful, and likable, with a strong passion for his research that he was able to transmit to his students.

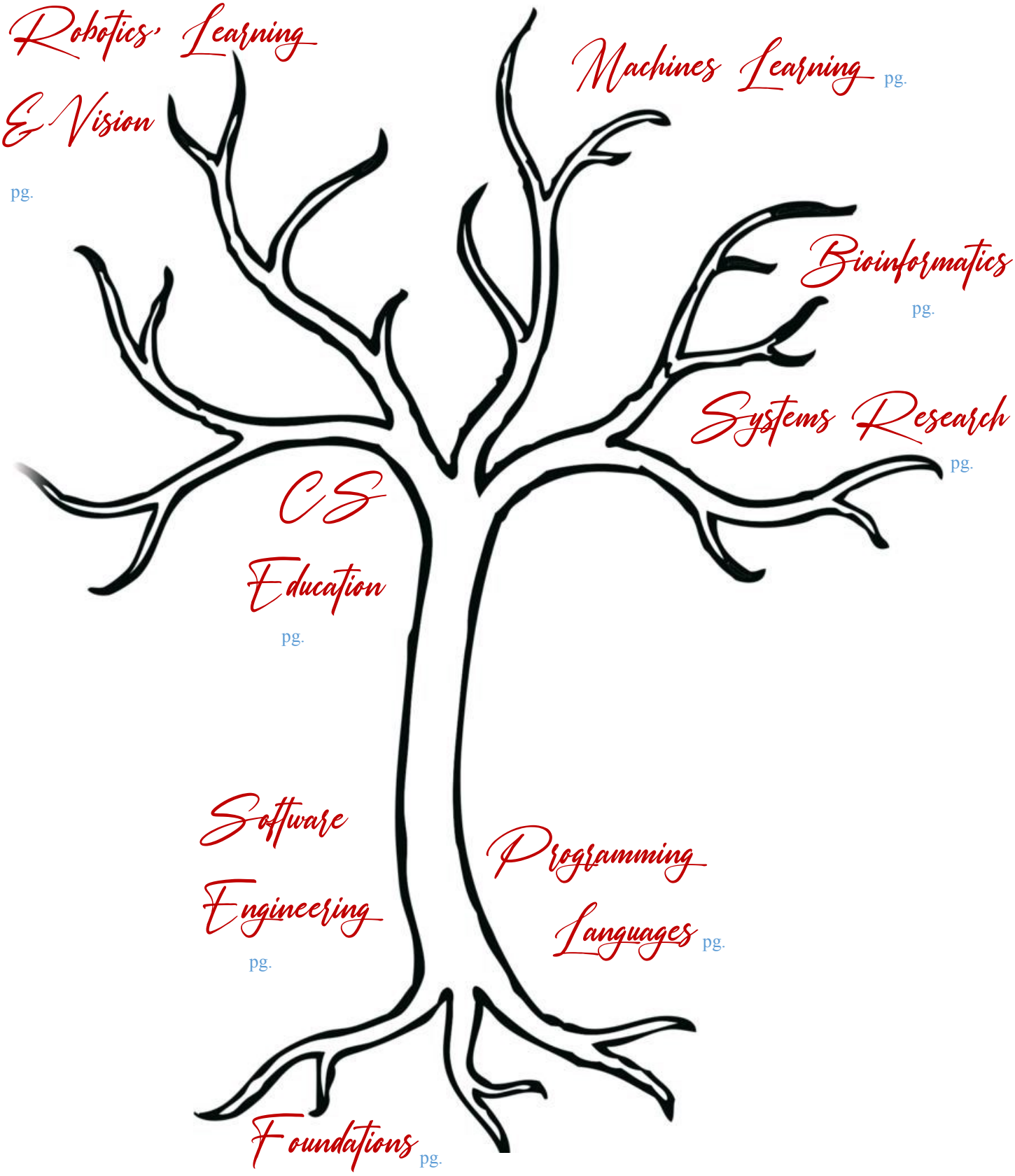
Professor Toussaint was a passionate teacher and an incredible mentor. He supervised sixteen Ph.D. students, eight of whom now hold tenured faculty positions at universities in Canada. In the words of a former student and colleague, “Godfried was a researcher who had an amazing ability to get people excited about problems. His flashes of intuition were spectacular and hard to resist. We all wanted to keep up and be part of this exciting adventure”. To the surprise of no one, Professor Toussiant received McGill’s David Thomson Award for excellence in graduate supervision and teaching. He has received many other awards and prizes for his body of work, among them the Senior Killam Research Fellowship by the Canada Council, the Canadian Image Processing and Pattern Recognition Society's Service Award for his work in computational geometry, and the Lifetime Achievement Award by CS-Can (the Canadian Association of Computer Science).

After retiring from McGill, Professor Toussaint had several appointments before settling down somewhat in 2011 as Professor and Program Head of Computer Science at New York University Abu Dhabi in the United Arab Emirates, a position he held until he died.

Toussaint loved sports and music. His pleasure in rhythms led him to become a good hand drummer, and some rumors claim that he was in the group that started the weekly drum circle by the George-Étienne Cartier Monument on Mount Royal. Also famous were the annual workshops on Computational Geometry that he organized at the Bellairs Institute in Barbados. They garnered an international reputation for being among the best collaborative research workshops in the world and have spawned similar workshops in several countries worldwide.



# Research @SOCS:A Family Tree



# Leading the way in Artificial Intelligence

**Key members of McGill's Reasoning & Learning Lab, which has built up an international reputation for its contributions to machine learning, are taking on important roles in the burgeoning area of A.I.**

**Artificial Intelligence** research goes back a long time at SOCS. **Monty Newborn**, who joined the School in 1975 and served as its director for eight years, was a pioneer in designing chess-playing and theorem-proving computer programs. He actually organized and officiated the famous Garry Kasparov vs DeepBlue matches in 1996 and 1997!


In the 1980s, **Renato De Mori** was among the first to study com- puter-aided speech recognition, which remains an important field of research to this day. Among Professor De Mori's many illustrious graduate students was a certain Yoshua Bengio, who would go on to earn a Turing Award for his work on neural networks.

A child's incredible learning ability is largely achieved through interaction with their environment and understanding what actions result in pleasant (getting their mother to smile) vs unpleasant (falling down) outcomes. **Doina Precup** is a pioneer of computational reinforcement learning (RL), where the same principle is used to teach robots or computer programs how to behave to maximize a certain reward. RL is today recognized as one of the major fields of machine learning. Professor Precup has established much of the has foundational theory of the field, while exploring a wide range of applications in robotics, and medicine, among other fields.

Any alum who passed through SOCS in the last 30 years will remember **Prakash Panangaden's** famously entertaining and challenging courses. Whereas Professor Panangaden's work has for a long time focused on fairly theoretical aspects of mathematics, physics, and programming languages, his more recent work looks into the very foundations of AI. Many of the most powerful approaches in modern AI are inherently probabilistic in nature – their success depends, to some extent, on the luck of the draw. Similarly, when robots operate in the real world, their observat-

ions about their environment are imperfect, and the way the environment evolves and consequences of their actions are somewhat unpredictable. Professor Panangaden lays the mathematical foundations to study the behavior of algorithms that rely on randomness, or that operate on non-deterministic data, with important consequences in RL.

**Joelle Pineau's** early work also involved the development of algorithms for reasoning about such probabilistic environments in robotics. Her more recent work has branched out in several directions, including dialogue systems and medical applications. For example, her team uses deep learning methods to automatically predict patient deterioration based on chest X-rays and other clinical variables and lab test results. Joelle was until 2020 President of the International Machine Learning Society, and she is a recipient of NSERC's highly prestigious E.W.R. Steacie Memorial Fellowship (2018).

Over the last few years, the School has greatly increased it's presence in AI research through the recruitment of several young stars. **Jackie Cheung** works in natural language processing and understanding (NLP/NLU), which enables computers to extract meaningful information from text written by and for humans, such as Wikipedia. His work integrates statistical machine learning models with structured databases of curated knowledge about the world to get AI systems to answer questions like, "Can a thought breathe a car?" Professor Cheung's team is developing systems that are trained to distinguish implausible events from plausible ones. Doing so will allow AI systems to better adapt to novel situations that they have not yet encountered.

Connecting language and mean- ing is also at the core of **Siva Reddy's** work, which aims to arm machines with strong language understanding abilities, such

that conversing with machines would feel as natural as conversing with humans. Through this process Professor Reddy hopes to discover fundamental representations of language. Recently he and his team won the Venturebeat AI Innovation of the year award in NLP, for his work on StereoSet measuring racism, sexism, and otherwise discriminatory behavior in a machine learning model.

The development of AI approaches that work to improve our society is also at the core of **Reihaneh Rabbany's** research. She develops computational approaches that combine machine learning and network science to help fight societal problems. For example, Professor Rabbany leads the "AI for Combating Human Trafficking in Canada (AICHTC)" project through which she develops techniques to detect organized activities in the online escort markets across Canada.

**Derek Ruths** also studies data that lives on networks for the purpose of detecting improper behavior. For example, his group develops data science and AI approaches for the detection of abusive speech in online platforms. Professor Ruths and his group are developing methods that use online communities that self-identify as hateful or as supportive of targeted abuse to create a more nuanced understanding of abusive language online.

Another burning societal issue our professors are helping to address is climate change. **David Rolnick**, who was recently listed among the "top 35 under 35" innovators by the MIT Tech Review, does foundational work in deep learning and inductive bias, as well as working to develop applications related to climate change mitigation, such as reducing greenhouse gas emissions, and adaptation, like responding to the effects of climate change.

We live in a geometric world, full of interesting symmetries that machine learning approaches can exploit to improve performance. **Siamak Ravanbakhsh** works in geometric machine learning, where he develops both the theoretical underpinnings of learning in a geometric world and investigates applications. An exciting application is toward the unsupervised anomaly detection methods that can identify a wide range of previously unknown anomalous patterns in large astronomical datasets.

Modern artificial intelligence (AI) relies on artificial neural networks, which are brain-inspired models of computation. However, current neural networks in AI are very restricted in their capabilities; they tend to do one thing well, but they are inflexible and unable to generalize to new conditions. **Blake Richard's** lab is working on new AI models inspired by an aspect of the brain that is not incorporated into existing neural networks in AI: episodic memories from specific events in our lives. Using simulations in 3D environments and comparisons to real recordings from human and animals brains he and his team are building AI systems with more human-like episodic memories, which are believed to be critical for our human ability to adapt to novel circumstances.

In 2017, McGill and UdeM joined forced to create Mila, the Quebec Institute for Artificial Intelligence. Today, the institute includes more than 31 core members, 12 of which are McGill associated, and 700 students. Mila serves not only as the epicenter for research in AI and machine learning, but its launch in the Mile-X neighborhood has contributed to attracting dozens of start-up companies and large research labs to Montreal, including but not limited to Facebook, DeepMind, Samsung, Google Brain, and Microsoft. SOCS professors are a major reason that Montreal is today such a hotbed of research in AI. In fact, many of our professors lead major industrial research labs in the city while continuing to work as McGill professors: Doina Precup is head of DeepMind Montreal; Joelle Pineau the Co-Managing Director at Facebook AI Research; Gregory Dudek and Xue Liu are Co-Directors of Samsung AI Center Montreal.

# Programming Languages, Software Engineering, and Education

Research in Programming Languages, Compilers, and Software Engineering at McGill investigates fundamental questions about how we can implement and maintain safe, reliable, and robust software.

Software is everywhere around us and our economy and society increasingly depend on it functioning correctly and efficiently. Failure and outages due to the malfunctioning of software incur billions of dollars in costs to our economy and can harm human lives. How to implement, validate, and maintain, software that runs efficiently is critical.

The McGill Software Technology Lab is jointly led by Professors **Jin Guo** and **Martin Robillard**. Members of the lab investigate how to facilitate the discovery and acquisition of crucial technical, design, and domain knowledge to support the development of complex software systems. Their research areas include automated documentation generation, software traceability, API usability, advanced programming tools, and computer science education. They work at the intersection of software engineering, artificial intelligence, and human-computer interaction.

The McGill Automated Reasoning, VErification, and Learning (MARVEL) group led by Dr. **Xujie Si** studies automated reasoning and learning techniques for building reliable software and intelligent systems. Their research is in the joint areas of formal methods, programming languages, and artificial intelligence. They develop machine learning based approaches to automate complex reasoning tasks like program analysis, synthesis, and verification. They also develop formal reasoning techniques to analyze machine learning robustness. Last but not least, they

explore a neuro-symbolic design to build interpretable and data efficient intelligent systems.

The Software Composition and Reuse (SCORE) Lab at McGill is jointly led by Professors **Jörg Kienzle** and Gunter Mussbacher. Their research areas include model-driven engineering, modularity, separation of concerns, reuse, software composition, software language engineering, and software engineering education.

In particular, they focus on Concern-Oriented Reuse (CORE), a next-generation software reuse paradigm in which software artifacts (models and/or code and their languages) are modularized by domains of abstraction within units of reuse called concerns. CORE combines techniques from software product line engineering, goal modeling, and software composition to create extremely versatile, generic units of reuse with well-defined interfaces that encapsulate a variety of solutions to recurring software development problems and allow the user to reason about the impact of selecting a solution on high-level stakeholder goals and system properties.

The Computation and Logic group led by Professor **Brigitte Pientka** investigates fundamental principles of programming languages -- their design and implementation -- to develop reliable, safe software. A central question that guides their research is how to design and implement safe, reliable programming languages that provide static guarantees about the programs that they write. To answer this question, they draw on logical frameworks and type theory, and exploit the deep and beautiful connection between proofs and programs. Their research areas include theory of programming languages (in particular functional programming languages), proof and programming environments, verification, and computer science education.

The Sable lab continues to explore different aspects of compiler-based program optimization. Led by **Clark Verbrugge**, current projects are focused on optimizing and extending WebAssembly, a relatively new, low-level language used to improve performance of compute-heavy web applications, and on optimizing GPU

compilation for specific application domains. Concurrent or parallel computation remains a theme, and is also part of a more theoretical investigation into the design and impact of abstract, "memory consistency" models, especially in relation to their impact on the ability to perform the gamut of compiler optimizations that drive program performance in modern, multi-threaded environments.

The Compilation and Architecture Synthesis Lab (CASL), led by Professor **Christophe Dubach**, conducts research at the intersection of software and hardware. They are witnessing a shift from general purpose processors towards specialized accelerators due to power constraint. However, designing and exploiting specialized accelerators is an extremely tedious and time-consuming task, often performed by hand. Given the rapidly changing nature of the application domain, the growing need for accelerators and the slow design process, this leads to a situation where tomorrow's hardware is already obsolete today. The lab focuses on developing automated techniques to generate and exploit accelerators which include: new data parallel programming abstractions to facilitate automation; new compilation techniques that map software onto specialized hardware automatically; and novel approaches to synthesize specialized hardware automatically from high-level abstractions. Overall, the research in the lab spans the entire system stack, from high-level language design to low-level hardware optimizations, including compiler optimizations.

How should computer science be taught to maximize the interest of the students while efficiently acquiring key concepts and skills? Computer science education is at the core of Professor **Elizabeth Patitsas'** research. But she doesn't limit herself to the purely pedagogical aspects of teaching; she also works on identifying ways in which technology can be harmful to particular groups of individuals, and how this can be mitigated. Professor Patitsas explores ways to change CS education to re-connect it with its social and historical contexts, through innovating pedagogy & curriculum, and supporting critical educators.

# Systems Research


System's research is a pillar of excellence in the School's research program, and covers a wide variety of topics, including Networking, Internet of Things, Cloud Computing, Cyber-Physical Systems, Green Computing, Large-scale Data management, Advanced Storage Systems, and Distributed Algorithms. Members of the Systems Group are **Oana Balmau**, **Joseph D'silva**, **Bettina Kemme**, **Xue Liu**, **Muthucumaru Maheswaran**, **Joseph Vybihal**.

As some example highlights of the last years, Prof. Liu and his students of the cyber-physical lab have worked on two impact research projects in the context of green computing: one aims at realizing energy-efficient, low-cost, and environment-friendly data centres that can use multiple energy sources and perform dynamic geographical load balancing across centres aiming at low energy costs, the other investigates how to best provide power for charging stations of electric vehicles across a region.

One of Prof. Maheswaran's project is looking at the many aspects of fog-computing where cloud resources are brought closer to the edge of the Internet and closer to the clients that need the services. The clients can be small-scale devices or mobile vehicles, making resource allocation and load-balancing challenging issues.

With key-value stored having become one of the predominant data storage systems of cloud systems, scalable and efficient solutions are crucial. Prof. Balmau is looking at how to create key-value stores that can take full advantage of the latest advancements in hardware technology, and are specifically designed to serve Data Science workloads.

In the AIDA project, spearheaded by Dr. D'silva and Prof. Kemme, data science computation is brought close to where much of the data resides i.e., within the database system, providing a unified programming interface to the users, and offering a unified approach to data analytics and management.



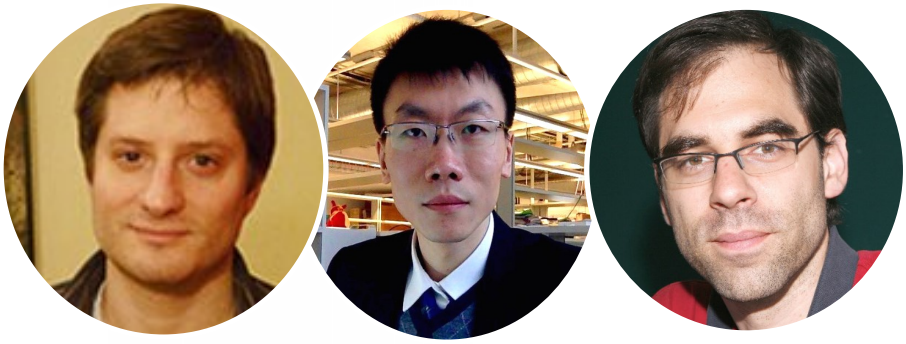
# Bioinformatics

In 2000, McGill was among the first universities to realize the need for the development of computational approaches for the analysis of biological data. Professor Michael Hallett launched the McGill Centre for Bioinformatics, soon joined by Profs. David Bryant and **Mathieu Blanchette**. While the former two have now left the university, the seed they planted is now strong.

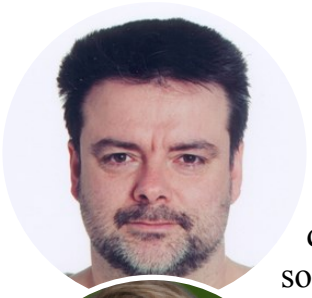
Genomics and molecular biology have seen a huge increase in the ability to generate large quantities of high-quality data about genetics and the inner working of cells. Prof. Mathieu Blanchette and his team are working hand-in-hand with biologists to develop computational approaches to make sense of this data.

Having recently joined McGill, Professor **Yue Li's** work aims to design statistical and machine learning approaches to study the genomics and biology of individual cells, and to link this information with the risk of developing diseases. MORE

RNA, the little cousin of DNA, is a type of molecules that is now recognized as a key player in a number of cellular processes and diseases. Professor **Jerome Waldispühl's** long-time interest has been on understand the structure of these molecules by combining algorithmic, statistical, and machine learning to understand RNA structure and function. He is also developing a completely unique approach to studying molecular biology problems – gamification! If, like millions of people worldwide, you've played the Borderlands 3 game, you probably encountered his Borderlands Science arcade-like game. Playing it actually helped an international team understand the human microbiome by improving the accuracy of RNA sequence alignment!



# Foundations



Computer scientists like to classify problems by how hard they are for computers to solve. On the one hand, showing problems are "easy" to solve (i.e. that there are algorithms that can produce the correct answer quickly) is the area of algorithm design, whose theory is deep and well-developed. Showing problems are hard to solve, on the other hand, is a much more difficult task since one must argue that *no* efficient algorithm for the problem exists. Professors **Robert** Robere and **Hamed Hatami** study complexity theory. They develop new tools to attack these "lower bound" problems by describing new equivalences between algorithms or types of proof systems. Of particular interest to these two professors is communication theory, which aims to capture a problem's complexity by measuring the minimal amount of information that two parties need to exchange (remember the famous Alice and Bob?) in order to jointly solve a given problem. Their work in this field, as well in combinatorial algorithms, is recognized worldwide.

Back to Alice and Bob – something else that they may want to do is communicate secretly over an unsafe channel (e.g. the internet); this is enabled by cryptography. But how can Alice be sure that she's really talking to Bob? And how can Bob prove his identity without then enabling others to fake it? Bob needs to demonstrate that he knows something that only he could know, but without revealing what that thing is – this is called a zero-knowledge proof. Professor **Claude** Crépeau works in developing zero-knowledge proof mechanisms that use and are robust against from quantum computers.

Often, the behaviour of algorithms or data structures depends on their input: they may be efficient on some inputs but slower on others. This leads to the natural question of what is the *distribution* of running times across all possible inputs, which is the field of study known as probabilistic analysis of algorithms. Professor **Luc Devroye** – one of the most recognizable figures in the department (if you've taken a course from him, you certainly remember him) – is the world authority in the field, as well as in other branches of probability theory.

Professor **Adrian Vetta** is our resident expert in a field that has rapidly increased in impact in the last few years: algorithmic game theory. This addresses questions that arise when algorithms are used in a competitive multi-agent setting. While the theory behind this has very deep roots (watch *A beautiful Mind*), the applications in economics are very real and practical: carbon credit trading, 5G frequency spectrum auctions, etc.

Algorithms used in both everyday life and science often need to solve complex numerical optimization problems: this is what happens when your GPS captures satellite signals to infer its position, and also when training an artificial neural network. Professor **Xiao-Wen Chang** focusses on obtaining efficient and robust numerical solutions to these types of highly practical mathematical questions.

# Robotics & Vision

The Robotics and Vision groups in the School of Computer Science work in some of the hottest research areas in the world today, and work closely with the Machine Learning community at McGill and elsewhere. Robotics and Vision is a long-standing area of exceptional strength at McGill, with the Center for Intelligent Machines (which we are part of) having been created back in 1983. Today McGill is also the lead site for the NSERC Canadian Robotics Network, Canada's only national organization for robotics research.

Robotics research in the School spans behaviour and gait learning, manipulation and the link between perception and action. Vision research enables computers to interpret images from cameras, as well as other sensors, such as medical imaging devices and ultrastructure imaging systems used in biology.

Several projects of Professor Gregory Dudek and Professor David Meger involve a combination of both analytical and learned models to produce control software for various kinds of autonomous vehicles. These vehicles can use cameras and other sensors to perceive the world and drive off road, including through the forest, along a dirt path, over the water surface, or even underwater. Having joined the School more recently, Professor Hsiu-Chin Lin is bringing expertise in control for locomotion and manipulation in complex, constrained environments. Progress on these fronts will allow vehicles to assist the forestry industry, perform search and rescue or do environmental monitoring. By combining different types of models in the control process, this work allows the vehicles to learn how to operate more efficiently and more robustly.

Professors **Kaleem Siddiqi** and **Michael Langer** work exclusively on vision, including the problem of 2D and 3D shape analysis and estimation, and depth perception. One set of problems involves modelling the geometry or appearance of 3D scenes and the image formation process as well as biological structures. For example, Professor Siddiqi's group has developed computational models of heart fiber modelling and estimation from diffusion MRI and microscopy. Both groups also explore fundamental connections between computer vision and human vision, with the goal of developing computational models of human vision.

Professor Paul Kry works on the opposite problem: how can we generate fake but physically realistic images and animation for video games, movies, and robotics simulations. Heading the Computer Animation and Interaction Capture Laboratory, an important aspect of his work is the combination of real-world measurements (e.g. motion capture), approximate models, and physically based simulations.


School of Computer Science