How British Columbia implemented its computer science education program

Brian Fowler
Yuri Nesen
Emiliana Vegas

April 2021

Center for Universal Education at BROOKINGS

How British Columbia implemented its computer science education program

Brian Fowler is a research analyst in the Center for Universal Education at Brookings.

Yuri Nesen is an intern in the Center for Universal Education at Brookings.

Emiliana Vegas is a senior fellow and co-director of the Center for Universal Education at Brookings.

Acknowledgements

The authors are grateful to Juliet Waters and Michael Hansen, whose comments on a previous draft greatly helped improve this version.

The Brookings Institution is a nonprofit organization devoted to independent research and policy solutions. Its mission is to conduct high-quality, independent research and, based on that research, to provide innovative, practical recommendations for policymakers and the public. The conclusions and recommendations of any Brookings publication are solely those of its author(s), and do not reflect the views of the Institution, its management, or its other scholars.

Brookings gratefully acknowledges the support provided by Amazon, Atlassian Foundation International, Google, and Microsoft.

Brookings recognizes that the value it provides is in its commitment to quality, independence, and impact. Activities supported by its donors reflect this commitment.

Summary

Computer Science (CS) education helps students acquire skills such as computational thinking, problem-solving, and collaboration and has been linked with higher rates of college enrollment (Brown & Brown, 2020; Salehi et al., 2020). A recent randomized control trial also showed that lessons in computational thinking improved student response inhibition, planning, and coding skills (Arfé et al., 2020). Since these skills take preeminence in the rapidly changing 21st century, CS education promises to significantly enhance student preparedness for the future of work and active citizenship.

CS education can also reduce skills inequality if education systems make a concerted effort to ensure that all students have equitable access to curricula that provide them with the needed breadth of skills—regardless of their gender, ethnicity, or socioeconomic status.

Based on prior analysis and expert consultation, we selected 11 country, state, and provincial CS education case studies that may apply broadly to other education systems. These cases have come from diverse global regions and circumstances and have implemented CS education programs for various periods and with different levels of success. As such, we have examined information to extract lessons that can lead to successful implementation in other parts of the world.

This study will focus on how the Canadian province of British Columbia (B.C.) developed its computer science program. B.C. is home to a growing technology industry and is one of the leading Canadian provinces for K-12 CS and computational thinking education (Gannon & Buteau, 2018). Its education system creates opportunities for students to pursue CS education based on their interests, including theoretical and quantitative concepts, practical applications, and problem-solving. Further, the province's approach to teacher professional development and student engagement merits consideration.

An overview of CS education in British Columbia

After decades of including computer literacy as part of the mandatory curriculum, B.C. announced its intention to begin instructing its K-12 students in CS in 2016 (Burgmann, 2016). District superintendents had expressed concern that teachers did not understand the topic well enough to teach it, and in response, the government invested 6 million Canadian dollars in professional development programs so teachers could include computational thinking concepts in their lesson plans. Between the summer of 2016 and the fall of 2017, the ministry contracted NGO partners Lighthouse Labs and Kids Code Jeunesse to create modules and deliver teacher training workshops across the province.

After two years of planning and preparation, implementation began in the fall of 2018. Today, the B.C. school system offers parallel electives in CS and programming as part of the Mathematics and Applied Design, Skills, and Technologies (ADST) curriculum for students in grades 11 and 12. These offerings appeal to students with both theoretical and creative inclinations and have the potential to generate interest in CS among secondary school students.

B.C.'s experience with CS education highlights four important lessons:

- Integrating CS into different areas of the curriculum can help make the subject more accessible to students based on their academic or professional inclinations.
- Incorporating marginalized groups' perspectives into pedagogy can foster inclusion in CS education. In B.C., select tribal schools have made strides in introducing First Nations and Indigenous People (FNIP) students to CS by integrating storytelling and games into CS concepts and activities.
- Universities play a critical role in creating a sustainable pipeline of preservice teachers who understand CS subject matter and pedagogy.
- Other stakeholders, such as NGOs and private companies, can play a significant role by lending technical expertise to providing professional development and certification opportunities for teachers focusing on CS.

Origins and motivation

Since the 1990s, B.C. had provided instruction in the use of information and communication technology (ICT) but did not teach more complex CS concepts. According to Jeffery, 2010, the provincial Ministry of Education had left the technology curriculum largely unchanged for the past two decades. Heading into the 2010 decade, few students understood the mechanisms that enable technologies to work, even though they knew how to use digital devices (Sherlock, 2016).

The origins of B.C.'s CS education trace back to calls from a growing technology industry (Doucette, 2016). In 2016, the province's technology sector employed 86,000 people—more than the mining, forestry, and oil and gas sectors combined, with high growth projections (Silcoff, 2016). The same year, leaders of the province's technology companies revealed in interviews that access to talent had become their biggest concern (KPMG, 2016). According to a 2016 B.C. Technology Association report, the province needed 12,500 more graduates in CS from tertiary institutions between 2015 and 2021 to fill unmet demand in the labor market (Orton, 2018). The economic justification for improving CS education in the province was clear.

In 2016, the B.C. government set out to recruit the Premiers Technology Council (PTC), an advisory board that aspires to make the province a global technology hub (T-Net, 2016). As part of the PTC's strategy, the B.C. Ministry of Education announced its plan to introduce CS in its curriculum. Premier Christy Clark said the intention was to give every student from K-12 the chance to learn the basics of coding and promised that "every secondary school boy and girl in B.C. will speak code by 2022" (Ma, 2017). In its K-12 ADST curriculum—a hands-on program that focuses on learning through design and creation in areas such as business education, ICT, and home economics—the ministry mandated computational thinking lessons for students at an early age to help reduce cognitive barriers when learning coding later on (Hamlin, 2016). As part of the announcement, the ministry pledged CA$6 million fund to support school districts that integrate CS into the curriculum.

Right after the announcement, ministry officials looked to Ontario and England as examples to follow. While officials noted that no education system had implemented

coding classes long enough to know the results (Burgmann, 2016), they thought that they could draw on other experiences for program design and implementation lessons.

By 2016, the provincial Ministry of Education, individual school districts, and nonprofit organizations had put together various teacher professional development activities to train in-service teachers in CS subject matter and pedagogy, including four days of dedicated workshops delivered in two phases. Today, the University of British Columbia (UBC) and University of Northern British Columbia offer courses to prepare preservice teachers for CS courses and ADST, including the computing component. Working simultaneously, these programs can train a group of teachers to fill the immediate need for CS teachers brought on by the mandatory curriculum while creating a steady pipeline of qualified teachers for the future.

The B.C. government implemented the new CS curriculum in a staggered approach. By 2017, the government made plans to introduce coding as a part of the 2018-2019 core curriculum (Julie, 2017). As part of the changes, the province's ADST curriculum was expanded with a greater focus on coding and computational thinking in grades 6 through 9 (Gannon & Buteau, 2018).

Currently, primary school students are required to learn computational thinking as part of the core curriculum. CS classes are mandatory from grades 6-9 and then become optional in high school as students get more of a chance to specialize. Throughout each grade level, students focus on theoretical and quantitative concepts using practical applications and problem-solving techniques.

Stakeholder involvement

Stakeholders—including private companies, universities, government organizations, and NGOs—have supported CS education in B.C. by organizing programs for youth and providing training to teachers. This section details these efforts.

With the help of the B.C. Ministry of Education and the British Columbia Institute of Technology (BCIT), Microsoft Philanthropies launched the Technology Education and Literacy in Schools (TEALS) B.C. program in 2018 to connect high school teachers to technology industry volunteers. The volunteer experts support instructors to learn CS independently over time and develop sustainable high school CS programs in B.C. Since 2018, the B.C. branch of TEALS has educated just over 400 students across four provincial high schools (Post-Secondary B.C.).

Drawing from their experience delivering the initial 2016-2017 teacher training workshops across B.C., Lighthouse Labs and Kids Code Jeunesse partnered in 2018 to launch "Code, Create, Teach," a workshop funded by the federal CanCode program that offered teachers in all Canadian provinces and territories coding skills that they can apply in the classroom (Orton, 2018).

Micro:bit, a nonprofit organization that makes and distributes small computers for educational purposes, has provided additional training sessions for school leaders on ways to incorporate computational thinking; in turn, B.C. schools have incorporated Micro:Bit circuit boards—designed for beginners in coding and technology—to introduce students to CS in an intuitive and accessible way.

Teacher preparedness, professional development, and certification

When CS was first announced in 2016, superintendents reported that most of their teachers were not prepared to teach the subject (Dolski, 2016). In response to this concern, Premier Clark announced the CA$6 million investment toward infrastructure and teacher training in CS, focusing on both subject matter and pedagogy (British Columbia Office of the Premier, 2016; Dolski, 2016). This included materials and "train the trainer" workshops developed by Light House Labs and Kids Code Jeunesse. Each school district was invited to send grades 6-9 teachers to regional, two-day training workshops that introduced teachers to computational thinking concepts and lesson planning for the content. Then, leading up to the 2018 CS education mandate, over 250 district leaders were given four-day training workshops in computational thinking and simple code.

Today, to familiarize themselves with CS, in-service teachers often participate in professional development workshops, where they carry out activities such as "human coding," in which one participant writes out instructions for a second participant, who reads out the instructions to a third participant to complete the task. Organizations including Science World B.C.'s Tech-Up, Canada Learning Code, and Cybera provide such workshops.

In addition to in-person professional development programs, district and nonprofit programs like CodeBC have given teachers online resources that can guide them in fulfilling the curriculum. They also encourage creative lesson plans like the use of CS Unplugged—analogue lesson plans that teach students computational thinking and basic programming skills. Microsoft TEALS also connects teachers with resources and experts, but is not an accredited institution and cannot issue course credits recognized in B.C. It can only "expect schools to work within district and state frameworks to appropriately acknowledge the teacher's achievement" as professional development credit hours, industry experience, or other forms of accreditation (TEALS, 2020).

B.C.'s university teacher education programs increasingly offer technology sub-specialties for preservice teachers. For example, UBC, the largest provincial university's teacher program, offered summer courses in the Computer Studies Education (CSED) and Technology Studies Education (TSED) areas of the secondary teacher education program since 1998. The courses focused on applications such as principles for engineering computers or robots (Krug, 2012). Despite such offerings, STEM education was not taught as a required subject at UBC until 2017. Before then, it was introduced into the teacher education program on an ad hoc basis as individual (i.e., stand-alone STEM course) or combined (i.e., math and engineering or science and technology courses) courses (Krug, 2012). UBC currently offers a technology education bachelor's degree program to prepare teachers for the ADST curriculum. CS candidates in UBC's teacher education program "prepare to teach ICTS–computer information systems, ICTP–computer programming, and ICTC–applied digital communications." B.C. educators must receive a certificate of qualification to work in public schools. To receive certification, teachers need at least 24 credits of post-secondary coursework in one of the listed subjected areas, which includes CS. Educators also need 48 credits from a recognized teacher education program that includes classroom teaching experience. Along with UBC, the University of Northern British Columbia offers CS training courses for preservice high school teachers, but does not have similar courses for elementary school teachers (Jeffery, 2010).

In September of 2020, the Ministry of Education piloted and previewed ShareEDBC, an invite-only platform available to all teachers and validated community partners to share resources, lesson plans, and best practices. Building on the communities of practice seeded in the "train the trainer" workshops, this platform provides an opportunity for teachers to share and take advantage of curated resources in computational thinking and code, alongside other subjects.

Curriculum and learning activities

B.C. students are exposed to computational thinking concepts as early as primary school, where they learn how to prototype, share, and test ideas. At the same time, the ministry gives teachers flexibility to tailor the curriculum to students' needs while working with the technology available to them (Dolski, 2016). In primary school, the B.C. curriculum emphasizes numeracy using technology and information technology. Students develop numeracy skills by using models and learn information technology skills to apply across subjects. From grades K to 1, objectives include presenting ideas using electronic documents. From grades 2 to 3, students should "demonstrate an awareness of ways in which people communicate, including the use of technology," in English language arts classes. They should also be able to find information using information technology tools. In grades 4 and 5, students focus more on prototyping and testing (Gannon & Buteau, 2018).

Middle school CS courses aim to help students understand the basics of coding with an opportunity to specialize in CS in later years (Silcoff, 2016). The grade 6-9 mandatory curriculum was designed so that schools can create their own content based on several available content modules, one of which is computational thinking. Grades 6-7 focus on programming languages and algorithms. Grade 8 teaches students debugging skills. In grade 9, coding is included in electronics and robotics, as well as ICT (Gannon & Buteau, 2018).

B.C. schools are required to offer two elective CS courses, each satisfying a mathematics credit requirement for graduation. The first course–CS 11, available for grade 11 students–emphasizes CS concepts of decomposition, algorithms, computational thinking, and problem-solving. The second–CS 12, available for grade 12 students–builds off the first course by providing more advanced instruction in the same topics. In both courses, students are expected to reason and model in situational contexts, as well as understand how to solve problems. They also learn to communicate and reflect on their thought process. Both courses integrate mathematical modeling as one of the learning goals, and CS 11 introduces practical applications of CS in financial analysis. Simultaneously, B.C. secondary schools offer computer programming

electives for students in grades 11 and 12 in ADST. According to the B.C. Ministry of Education, the ADST curriculum "builds on students' natural curiosity, inventiveness, and desire to create and work in practical ways" and includes skills and concepts from disciplines such as home economics, business education, information technology, and technology education.

While the CS 11 and 12 electives place greater emphasis on mathematical modeling, the computer programming grades 11 and 12 electives in ADST encourage students to design a solution to a problem and manage a project as they complete different steps in their design cycles. As part of a design cycle, students understand the context of a problem through user-centered research, define a design opportunity, imagine, create prototypes, test their work, use project management processes for production, and share their progress in the design cycle.

Box 1. A timeline for the development of computer science education in British Columbia

2011 – The B.C. initiative ANCESTOR pilots computer education for indigenous students in the Lau Welnew Tribal School

2016 – B.C. premier announces plan to teach CS across all grade levels in all schools

2016-2017 – the Ministry of Education offers teacher training workshops to district leads from six regions across the province

June 2018 – The provincial Ministry of Education announces CA$6 million investment in teacher training

September 2018 – Schools begin to fully implement CS education reform

Inclusion

The B.C. Ministry of Education and other stakeholders have taken steps to give girls, women, and FNIP students a chance to develop an interest in CS education. This includes research on new pedagogies, mentoring programs, and summer camps.

After-school programs have taken specific steps to increase girls' participation in CS education. The UBC Department of Computer Science runs GIRLsmarts4tech, a program that focuses on giving girls in grade 7 role models and mentors that encourage them to pursue technology-related interests. Companies such as Microsoft and SAP also support CS education equality. Microsoft supports the YMCA summer day camp that hosted Ladies Learning Code and hosts workshops that help girls and women develop digital skills (Sherlock, 2016).

Crucially, part of the B.C. curriculum for C.S. education seeks to incorporate FNIP world views, perspectives, knowledge, and practices with CS concepts. This is important, since as of 2016, B.C.'s FNIP population—including First Nations, Metis, and Inuits—was 270,585, up 38% from 2006. With 42.5% of the FNIP population under 25, it remains critical for the province to deliver quality education to this young and growing group (Ministry of Advanced Education, Skills and Training, 2018).

Research on pedagogy finds that FNIP students in the region value stories, prefer collaborative over competitive learning, and learn best through imitation, observation, and trial-and-error. In line with these findings, the Lau Welnew Tribal School near Victoria, B.C. piloted a computer curriculum in 2011 based on "digital storytelling," which is a "narrative expressed in digital form for a variety of purposes, with applications ranging from education to personal expression, record keeping to movement promotion and everything in between" (Westor & Binn, 2012). Adaptations to the curriculum were then delivered to a summer camp for the Songhees First Nation in 2012 and a fall 2012 elective at Shoreline

Middle School in Victoria, B.C. (Camosun, 2011). The B.C.-based ANCESTOR project (AborigiNal Computer Education through STORytelling) has also organized courses and workshops to encourage FNIP students to develop computer games or animated stories related to their culture and land (Westor & Binn, 2012). Students learned to create animations using Carnegie Mellon University's Alice, a 3D programming software with a drag-and-drop interface (Westor & Binn, 2012).

Conclusion

In 2016, B.C. set a goal to implement coding across all grade levels in its education system and enhance CS instruction in primary and secondary schools. The province purposely included primary school students because it understood that students who have early exposure to computational thinking and CS will face fewer learning barriers as they advance to higher grades.

Civil society organizations such as Microsoft Philanthropies have enhanced teacher professional development and certification by offering accreditation to teachers on a district level as part of initiatives such as the TEALS program. Additionally, these organizations, including UBC, have played a significant role in expanding CS access to women and girls by hosting workshops and other initiatives. NGOs such as ANCESTOR have also introduced FNIP students to CS by leveraging pedagogy tailored to indigenous students. Students in tribal schools have engaged with CS by developing their own games based on stories using visual programs such as Alice.

B.C. secondary schools offer CS electives in both the mathematics and ADST areas of learning. These offerings create opportunities for students to pursue CS education based on their interests, whether they are more interested in theoretical and quantitative concepts or practical applications and problem-solving.

References

$6 million to help connect students with coding new curriculum and computers [Press release]. (2016, June 10). Retrieved from https://www.sd68.bc.ca/wp-content/uploads/MoE-Coding-and-New-Curriculum-June-10-2016.pdf.

2016 BC technology report card. (2018, February 1). Retrieved March 02, 2021, from https://wearebctech.com/2016-bc-technology-report-card/.

Aboriginal Learners in British Columbia's Public Post-Secondary System. (2018). Retrieved from https://files.eric.ed.gov/fulltext/ED590249.pdf.

BBC micro:bit. (n.d.). Retrieved March 02, 2021, from https://bc-robotics.com/product-category/bbc-microbit/.

BCIT and Microsoft launch technology education program for high schools: Post-Secondary BC. (2019, January 17). Retrieved March 2, 2021, from https://www.postsecondarybc.ca/news/bcit-and-microsoft-launch-technology-education-program-for-high-schools/.

BC's Curriculum: Computer Programming 11. (2021). Retrieved March 02, 2021, from https://curriculum.gov.bc.ca/curriculum/adst/11/computer-programming.

BC's Curriculum: Computer Science 12. (2021). Retrieved March 02, 2021, from https://curriculum.gov.bc.ca/curriculum/mathematics/12/computer-science.

British Columbia, Canada, Ministry of Advanced Education, Skills, and Training. (2018, June). Aboriginal Learners in British Columbia's Public Post-Secondary System. Retrieved from https://files.eric.ed.gov/fulltext/ED590249.pdf.

British Columbia, Canada, Ministry of Education. (2016). Taking Making into Classrooms. Retrieved from https://mytrainingbc.ca/maker/en/toolkit/Taking_Making_into_Classrooms.pdf.

Burgmann, T. (2016, January 19). B.C. adds coding to K-12 school curriculum. Retrieved from https://www.ctvnews.ca/politics/b-c-adds-coding-to-k-12-school-curriculum-1.2742593.

Burgmann, T. (2016, January 19). B.C. adds coding to K-12 school curriculum. Retrieved March 02, 2021, from https://www.ctvnews.ca/politics/b-c-adds-coding-to-k-12-school-curriculum-1.2742593.

CanCode. (2019, November 22). Retrieved March 2, 2021, from https://www.ic.gc.ca/eic/site/121.nsf/eng/home.

Chapman, M. (2000). Government of British Columbia (British Columbia, Canada, Ministry of Education). Retrieved from https://www2.gov.bc.ca/assets/gov/education/early-learning/teach/resources/primary_prog.pdf.

Coding coming to B.C. schools. (2016, June 12). Retrieved March 02, 2021, from https://www.cbc.ca/news/canada/british-columbia/coding-education-school-computers-1.3631720.

Computer science & technology education program. (n.d.). Retrieved March 02, 2021, from https://www.microsoft.com/en-us/teals.

CS Unplugged. Computer Science without a Computer. Retrieved March 2, 2021, from: https://csunplugged.org/en/.

Dolski, M. (2016, September 1). New school curriculum will teach B.C. students computer coding. Retrieved March 2, 2021, from https://www.theglobeandmail.com/news/british-columbia/new-school-curriculum-will-teach-bc-students-computer-coding/article31686610/.

Doucette, K. (2016, May 06). Racing 'beebots' to teach kids coding in Nova Scotia | CBC News. Retrieved March 2, 2021, from https://www.cbc.ca/news/canada/nova-scotia/beebots-mcneil-casey-coding-programming-education-computer-1.3571568.

Gannon, S., & Buteau, C. (2018, April). Integration of Computational Thinking in Canadian Provinces [Scholarly project]. Retrieved from http://ctmath.ca/wp-content/uploads/2018/10/Symposium_CanadaMap_Gannon-Buteau.pdf.

GIRLsmarts4tech: Computer science workshops for girls. (n.d.). Retrieved March 02, 2021, from https://www.cs.ubc.ca/girlsmarts4tech/.

Hamlin, J. (2016, October 22). BC's new "coding" curriculum. Retrieved March 02, 2021, from https://cuebc.ca/2016/01/21/bcs-new-coding-curriculum/.

Introduction to Applied Design, Skills, and Technologies. (n.d.). Retrieved March 2, 2021, from https://curriculum.gov.bc.ca/content/applied-design-skills-and-technologiesintroduction#:~:text=The%20Applied%20Design%2C%20Skills%2C%20and,Economics%20and%20Culinary%20Arts%2C%20Information.

Jeffery, J. (2010). Breaking the Mold: Why Computer Science Needs To Be a Fundamental Science Within the BC Curriculum [Scholarly project].

Julie, A. (2017, August 28). Teaching coding in Canadian schools: How do the provinces measure up? Retrieved March 2, 2021, from https://globalnews.ca/news/3693932/teaching-coding-in-canadian-schools-how-do-the-provinces-measure-up/.

KPMG (2016). British Columbia technology report card: Scaling up BC's tech ecosystem. https://wearebctech.com/2016-bc-technology-report-card/.

Krug, D. H. (2012). STEM education and sustainability in Canada and the United States. In *2nd international STEM in Education conference, Beijing, China.* http://stem2012.bnu.edu.cn/data/long%20paper/stem2012_87.pdf.

Ma, M. (2017, March 16). Opinion: From no computers to Alternative Facts: Coding in B.C. schools after one year. Retrieved March 02, 2021, from https://vancouversun.com/opinion/opinion-from-no-computers-to-alternative-facts-coding-in-b-c-schools-after-one-year.

Microsoft TEALS Implementation Guide (2021-22 School Year). (n.d.). Retrieved from https://www.tealsk12.org/wp-content/uploads/2020/10/2021-22-TEALS-Implementation-Guide.pdf.

Nine new members appointed TO PREMIER'S Technology Council. (2021, March 02). Retrieved March 02, 2021, from https://www.bctechnology.com/news/2016/12/12/Nine-New-Members-Appointed-to-Premiers-Technology-Council.cfm.

Orton, T. (2018, November 16). Kids' coding programs target B.C. tech Talent gaps. Retrieved March 02, 2021, from https://biv.com/article/2018/11/kids-coding-programs-target-bc-tech-talent-gaps.

Our Micro:bit Story - School District 43 – Coquitlam, B.C., Canada [Brochure]. (n.d.). Retrieved from https://www.sd43.bc.ca/Resources/EdTech/Documents/OurMicrobitStory.SD43.pdf.

ShareEdBC. (n.d.). Retrieved March 2, 2021, from https://www.shareedbc.ca/.

Sherlock, T. (2016, September 3). Computing the value of coding ahead of its introduction to b.c. curriculum. Retrieved from https://vancouversun.com/news/local-news/computing-the-value-of-coding-ahead-of-its-introduction-to-b-c-curriculum.

Technology education. (n.d.). Retrieved March 2, 2021, from https://teach.educ.ubc.ca/technology-education/.

To inspire aboriginal youth: Computer skills spark cultural storytelling. (2011, February 2). Retrieved March 2, 2021, from

http://camosun.ca/news/old/2011/feb/aboriginal-youth.html?reroute_notice=archive.

The University of British Columbia - Grades K-12 | Computer Science. (2018, March 28). Retrieved March 02, 2021, from https://www.cs.ubc.ca/grades-k-12.

What is policy? (2021, January 25). Retrieved March 02, 2021, from https://www2.gov.bc.ca/gov/content/education-training/k-12/administration/legislation-policy/what-is-policy.