

Coding and Robotics in the Elementary Curriculum Overview

The twenty-first century has brought us many wonderful changes to our world including access to excellent educational technology. This comes in many forms from mobile phones in our students' pockets to programmable drones flying through the classroom door, with each offering opportunities to engage learners in new ways.

This resource jumps into this space and offers educators ideas and guidance to use technologies in their classrooms.

Each idea that is presented in this project offers low tech (or unplugged activities) as well as high tech (plugged activities) options with a particular focus on Coding and Robotics platforms. As you read through each resource, it should be clear how seamlessly both inquiry and technology can be added to your program in truly meaningful ways.

This resource has contemplated how technologies like coding and robotics can be used to illuminate the expectations and big ideas in the Ontario Science and Technology Curriculum 1-8. Throughout it, we have referenced many terms and technologies, and what follows is meant to define these for you and provide background and locations for further inquiry.

What is Coding?

- A set of step-by-step Instructions that tells a website, app, cell phone, game, software or robot what to do
- Each instruction needs to be detailed and specific enough for the device or program to understand and carry it out
- If the instructions are not correct, the program will not work the way you want it to. Debugging the code helps to fix the problem
- Many of the devices and gadgets we use have a program or a code that has been written by a programmer to make it run effectively
- Computers and other devices can only follow instructions in programming languages they understand. A programmer selects the language that is most suitable for the task
- There are a number of different programming languages. For example, Scratch is a visual language that is ideal for learning programming
- Coding forms the basis of modern technology that is used in our schools and the workplace

Curriculum Connections

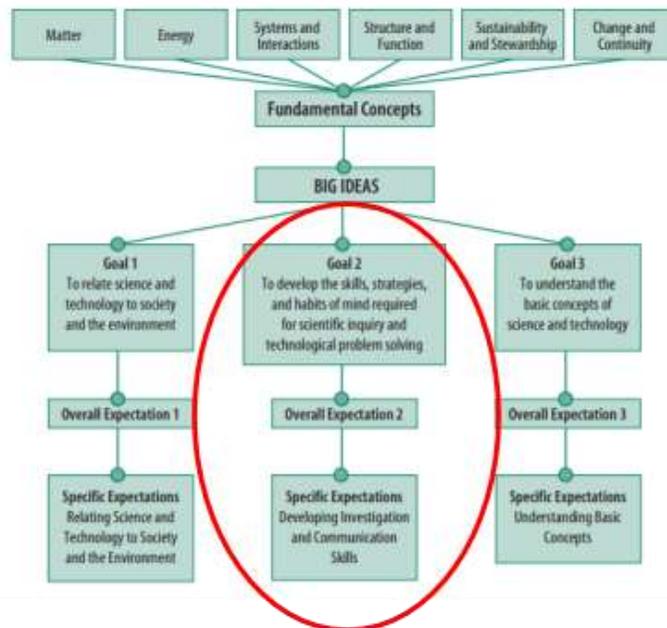
If you are looking to add coding and robotics to your classroom, but aren't sure how it directly connects to the curriculum to support your usage, look no further. You may not be able to open a PDF of any Ontario Curriculum document, hit Ctrl + F and find direct references to coding, but that doesn't mean coding can't be a meaningful part of your program.

There are four main gateways to embedding coding within the Ontario Curriculum:

- 1) Technological Problem-Solving
- 2) Computational Thinking
- 3) Inquiry Process
- 4) Creative Application to Many Curriculum Points

1. Technological Problem-Solving

The Ontario Science Curriculum has three primary goals embedded within every strand and grade. Goal Two, which states the necessity “to develop the skills, strategies, and habits of mind required for scientific inquiry and technological problem-solving” (MOE, p.3) is a natural fit for any coding activities you wish to explore. Technological problem-solving can be found as a Specific Expectation from either 2.2 to 2.6 across all grade levels within the curriculum.



(MOE, p. 3)

The main criteria of the Technological Problem-Solving Continuum are, “testing, retesting, and modifications of a product or process; communicating about the solution; and recommending of changes or improvements,” (MOE, p.16) which are critical components of working with code or robotics. When implementing these explorations, you can use the detailed continuum found on page 17 of the Science and Technology Curriculum to build your success criteria and learning goals with a clear curricular connection.

2. Computational Thinking: Why is it Important?

Coding is simply the process of choosing the correct computational strategy to provide a computer with the instructions necessary to complete a task. The difference with coding is that the outcome is usually highly motivating, as it takes the form of an animation, online game or completion of a mechanical task by a robot, which can elicit cheers of joys from the creators.

As stated in [Computational Thinking: Why is it Important?](#), computational thinking allows students to take a large, difficult problem and break it down into several simpler problems, to solve it more efficiently using decomposition, pattern recognition, abstraction, and algorithms.

The creation of code, analysis of code, and debugging of code is a natural exploration of these steps.

Problem Solving is also one of the seven major processes outlined in the Ontario Math Curriculum. Coding and robotics can reinforce or even introduce the Problem Solving Model.

Education is changing from being teacher-centred with a high focus on fact memorization, to student-driven where students are encouraged to be innovators, creators, and critical thinkers. Our role as educators is to facilitate deep learning by exposing students to real world problems so that they are challenged to think outside the box in order to find solutions. In many of our schools, coding has become a means by which students can challenge themselves to become critical thinkers and problem solve.

Figure 1: A Problem-Solving Model

Understand the Problem (the exploratory stage)
<ul style="list-style-type: none">➤ reread and restate the problem➤ identify the information given and the information that needs to be determined <p><i>Communication:</i> talk about the problem to understand it better</p>
Make a Plan
<ul style="list-style-type: none">➤ relate the problem to similar problems solved in the past➤ consider possible strategies➤ select a strategy or a combination of strategies <p><i>Communication:</i> discuss ideas with others to clarify which strategy or strategies would work best</p>
Carry Out the Plan
<ul style="list-style-type: none">➤ execute the chosen strategy➤ do the necessary calculations➤ monitor success➤ revise or apply different strategies as necessary <p><i>Communication:</i></p> <ul style="list-style-type: none">➤ draw pictures; use manipulatives to represent interim results➤ use words and symbols to represent the steps in carrying out the plan or doing the calculations➤ share results of computer or calculator operations
Look Back at the Solution
<ul style="list-style-type: none">➤ check the reasonableness of the answer➤ review the method used: Did it make sense? Is there a better way to approach the problem?➤ consider extensions or variations <p><i>Communication:</i> describe how the solution was reached, using the most suitable format, and explain the solution</p>

Computational thinking allows students to take a large, difficult problem and break it down into several simpler problems, in order to solve it more effectively. There are four major parts to computational thinking: decomposition, pattern recognition, abstraction, and algorithm.

Decomposition: breaking down the problem into its basic parts, figuring out what all the parts are, and examining how a task might be divided up through chunking

Pattern Recognition: finding the similarities and differences between the different parts in order to help make predictions

Abstraction: finding the general principles that help find the patterns and looking for the patterns that do not fit and removing them

Algorithm: developing the step- by-step instructions that solve similar problems

When students use computational thinking to help them solve problems, they become more detailed orientated and are better able to break down a problem. This allows them to communicate their thoughts and ideas more effectively.

3. Inquiry Process

Inquiry-Based Learning is an important 21st Century Learning model and is referenced in every Ontario curriculum document, including the Together for Learning document put together by the Ontario School Library Association in association with the Ministry of Education.



Embedding coding and robotics into this process is simple. From the exploration of the workings of a mechanism to the use of programs such as Scratch to create a finished product, you can find an entry point through Inquiry. In fact, every project in this resource is inquiry-based with the goal of engaging students and allowing for differentiated explorations and products. Coding moves our students from the status of content consumers to content creators.

4. Creative Application to Many Curriculum Points

Our application of coding and robotics is only limited by our imagination. Many specific outcomes can be hit depending on how you engage with coding and the curriculum.

The Toronto District School Bo has completed extensive work in this area. If you go to their [Introduction to Coding in the Elementary Grades](#) and click on [TDSB Coding Curricular Connections with Scratch](#) you will find Strands, Overall Expectations, and Specific Expectations across The Arts, Science & Technology, Mathematics, and Language Curriculum Documents for Grades 4 to 8 partnered specifically with tasks to be completed in the coding creation and exploration site, [Scratch](#).

A key researcher in the area of coding and the curriculum, specifically in the area of mathematics, is Lisa Anne Floyd. She focuses on Computational Thinking in her teaching at Western University and teaches Computer Science and Mathematics in the Thames Valley District Board and works to promote the importance of coding in classrooms across Ontario. Her site, [Integrating Coding Into the Elementary School Curriculum](#), has many activities and resources and is sorted by subject and strand.

SPICE model

Most of the coding and robotics activities are design challenges that follow the technological design process in the curriculum document. To simplify the steps for the process, we decided to follow a **SPICE** template (**Scenario, Problem, Investigation, Construction, Evaluation**).

The **scenario** is explained on a separate slide. Typically the teacher chooses a scenario, a hook, a situation. The ideas can come from a book, news story, scientific article or a situation in the school or surrounding community. For example, a scenario could be : *"You have been hired to design a new amusement ride for a grand celebration. You will build a working (moving) model of a safe carnival ride."*

From the scenario, the teacher or the class or students by themselves can define a **problem**: what do they need to design and construct exactly and what are the criteria for success (how will they know if their design is successful). For the previous scenario, a specific problem could be a specific choice of ride and the criteria for success is what the ride needs to do specifically. For example: *a Ferris wheel that needs to turn at a safe speed, stop to let people in and out, and be stable*. Though the teacher can direct students with choosing a problem, the criteria for success should be developed with the student and not given to them directly for a more powerful experience. Also, if the design is for another person / a client, the student can interview the client for what they feel their needs are to add to the criteria for success; this may be related to special functions or aesthetics.

The **investigation** part of the design can be teacher-led with mini-lessons or experience to touch key concepts and skills needed for the design and construction of the solution to the problem. They can

also be student-led where students are experimenting with various materials for building, or various designs.

Once students have all the information they need and they have reviewed their design, they **construct** or code their design. In some cases, a full build is not possible because of constraints (time, space, materials, safety), in which case a clean copy of the drawn design, or a word explanation of the code, could be presented.

In terms of **evaluation**, we mean that students are evaluating their design according to the criteria for success they had set for themselves at the beginning of the activity. This is not the teacher evaluation, though the teacher may choose this metacognition exercise to inform learning.

DESIGNING WITH SPICE



Engineering Design Process

1. Ask:

- At this stage, teachers are leading students through the inquiry process. Present students with a relevant problem whether it be something that is happening in the city they live in or something that a character in a book is dealing with. Students will learn to define the problem and do the preliminary research on past solutions. They will then gain a deeper understanding about the existing problem.

2. Imagine:

- Students will then start to imagine and brainstorm different solutions. Students then define the idea they like the best.

3. Plan:

- Students start to draw and design a solution to the problem. They start to draw blueprints and decide what materials will be used for their prototype. While planning, it is important to encourage students to define what their criteria for success is. How will they know that their prototype, design or model is successful?

4. Create:

- The fun part! Students start to bring their plans to life. They use the materials and create the actual prototype based on their plans.

5. Test:

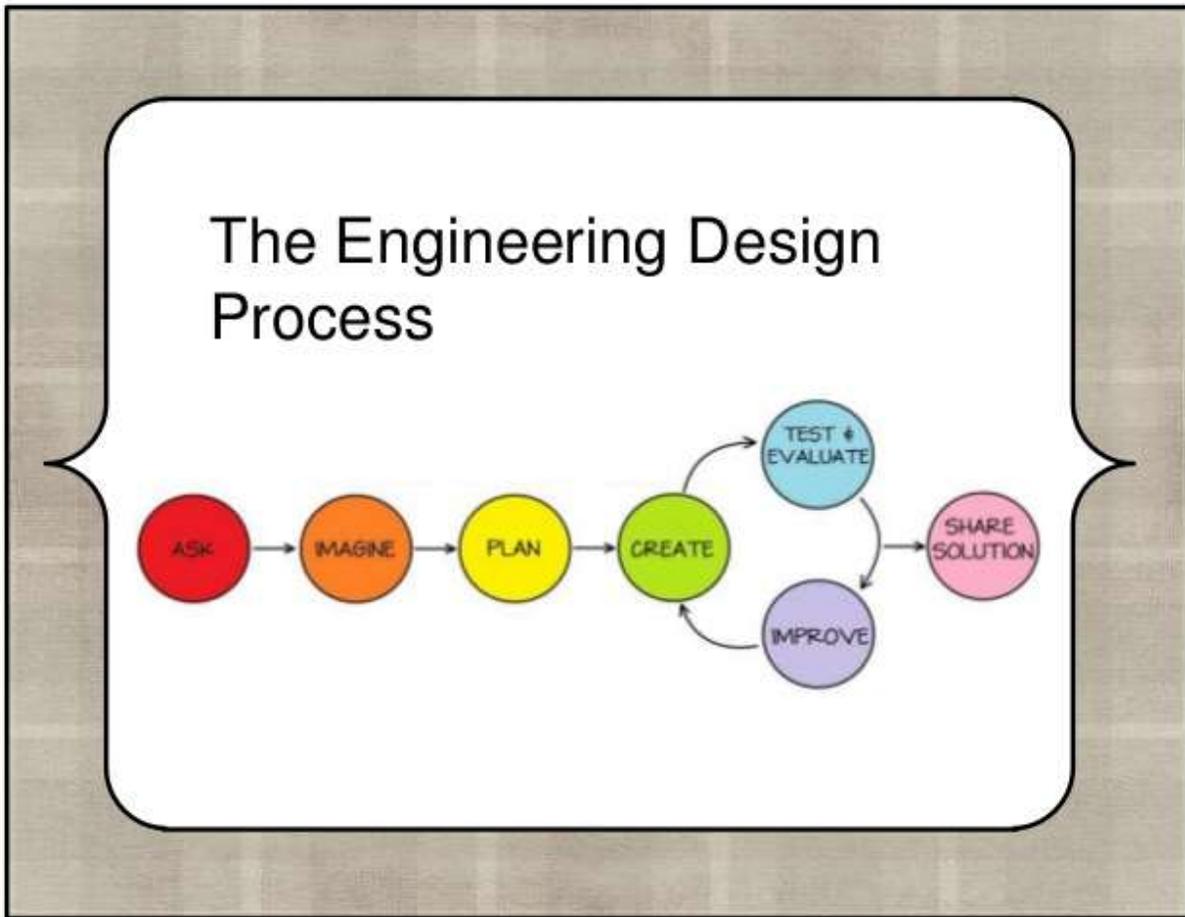
- As they are creating, students will continuously be testing their design. As they start building, students may start to realize that many things did not go as they had originally planned.

6. Improve:

- After their test, students will define how they can improve their design based on their failures. How can they redesign their prototype so that it can reach success?

7. Share:

- It is important to give students the ownership to share their process and their learning! Students have the opportunity to develop their entrepreneurial and leadership skills by showcasing their learning and their product. This also gives other students an opportunity to learn from their peers.



Why Create Scenarios for Design Problems?

When developing a design problem with and for students, it is important to ground the problem in a scenario. This achieves two goals. (1) It provides a context for students to make connections to real life situations and their own experiences and (2) it poses a problem to solve that has real life implications.

(1) When students can make connections to real life situations and their own experiences, it provides opportunities for teachers to teach in a more transdisciplinary approach. Teachers can make connections to different subject areas through a design challenge. This provides a rich learning experience for students.

(2) Including a design challenge with real life implications ensures that students have a reason to test their design, evaluate whether it is successful, and redesign if necessary. As a result, students have opportunities to engage in the Engineering Design Process.

Consider the following example that compares two different approaches to introducing a design challenge:

- 1) Build a bridge.
- 2) Stockholm, Sweden is spread across 17 islands. It has a population of approximately 3.6 million people. Currently, the bridges that connect the rural parts of the city to the downtown urban parts of the city cause extreme traffic delays. Design and build a bridge that connects the islands together and reduces traffic congestion.

While example 1 provides a building challenge for students, it is not set in a scenario and, therefore, does not provide students an opportunity to make rich and meaningful connections to the problem. In comparison, example 2 is grounded in a real life scenario to which students can make connections. It encourages students to use many of the 21st century skills such as researching, planning, testing, redesigning, and collaborating.

How to Create Scenarios

There are many sources for a design problem scenario.

Picture books

- Picture books provide a wide range of fiction-based or real life scenarios that can be presented in an engaging way.
- Fiction books can be read halfway through when a problem is introduced. This problem can then be used as a scenario for a design challenge.

Real life events

- Read newspapers, watch the local, national, or global news and identify an issue that can be used as a scenario for a design challenge.
- The Toronto Islands are flooded. How can we empower students to design a solution for this problem?

Students' interests

- Pay attention to what your students are interested in. Engagement is very important in learning. If your students are already interested in a topic or a problem that they have identified, use that as your source of inspiration for a design problem scenario.
- Fidget spinners have taken over your classroom? How can you use that excitement and engagement to your advantage? How can you create a design challenge that involves fidget spinners?

To frame your design challenge, use the following framework:

- 1) Explain the situation and state the facts.
- 2) State the current solution and identify the problem with it OR identify the problem.
- 3) Pose the design problem and consider how students will know that they are successful. (What does the bridge need to do?)

References

Blogger, G. (2016). *What is Computational Thinking and Why is It Important? A New Role for Students* — Megan Brannon, Garaway Local Schools. Ohio Department of Education. Retrieved 16 July 2017, from <http://tpts://education.ohio.gov/Media/Extra-Credit-Blog/November-2016/GUEST-BLOG-What-is-Computational-Thinking-and-Why>

Goal 2: Science Curriculum Flow Chart. [Image, screen capture]. Retrieved from <http://www.edu.gov.on.ca/eng/curriculum/elementary/scientec18currb.pdf>

Figure A: A Problem-Solving Model. [Image, screen capture]. Retrieved from <http://www.edu.gov.on.ca/eng/curriculum/elementary/math18curr.pdf>

4-part Research Model. [Image, screen capture]. Retrieved from <http://library.queensu.ca/research/guide/education-inquiry/model>

Computational Thinking: Why is it Important? [Google Doc] Retrieved from <https://docs.google.com/document/d/1PZ9DmBg9jMIGbczMHckauoZ0D172-PxvOLfw5TI5YyE/edit>

Loo, Wayne., Schmitt, Andrew., Millan, Julie., Lowry, Shelley., Bradbeer, Kevin. (n.d.) . *Introduction to coding in the elementary Grades* [Website]. Retrieved from <https://sites.google.com/a/tdsb.on.ca/dll/workshop-resources/getting-started-with-coding>

Loo, Wayne., Schmitt, Andrew., Millan, Julie., Lowry, Shelley., Bradbeer, Kevin. (n.d.) . *TDSB Scratch Coding Connections* [Google Doc]. Retrieved from https://docs.google.com/document/d/1zNGHxZCWesO-wSMh10l-ruC2axP1E-Q_dfpcVISdB2U/edit#heading=h.y0trnaxlyh3p

Ontario Ministry of Education. (2005). *The Ontario Curriculum Grades 1-8: Mathematics* [Program of Studies]. Retrieved from <http://www.edu.gov.on.ca/eng/curriculum/elementary/math18curr.pdf>

Ontario Ministry of Education. (2007). *The Ontario Curriculum Grades 1-8: Science and Technology* [Program of Studies]. Retrieved from <http://www.edu.gov.on.ca/eng/curriculum/elementary/scientec18currb.pdf>

Ontario Ministry of Education. (2010). Growing Success: Assessment, Evaluation, and Reporting in Ontario's Schools, First Edition Covering Grades 1-12.

<http://www.edu.gov.on.ca/eng/policyfunding/growSuccess.pdf>

Ontario School Library Association. (2010). *Together for Learning: School Librarian and the Emergence of the Learning Commons*. [Program of Studies]. Retrieved from

<https://www.accessola.org/web/Documents/OLA/Divisions/OSLA/TogetherforLearning.pdf>

Queen's Library. (2014, September 11). Education: Inquiry : Ontario Inquiry Model K-12 [Website]

Retrieved from <http://library.queensu.ca/research/guide/education-inquiry/model>

TELO. (n.d.). *Coding in Elementary: A Professional Learning Resource for Ontario Educators..* [Website].

Retrieved from <http://www.edugains.ca/resourcesTELO/CE/CodingInterface/interface.html>