

This resource was developed with funding from the Government of Canada CanCode program. We are required to collect participant data; it helps the government administrators understand the breadth of access and the impact of these kinds of programs.

If you are using this resource with learners and have received it from a source other than by participating in a Science World online program please fill out our participant survey <http://tinyurl.com/techuprequest>

Coding Math

Learning Kit (Gr. 4-6)



All classes are running differently right now. Our team is available to collaborate with you to adapt this for your classroom needs. Please do not hesitate to email us at techup@scienceworld.ca for any support you need with this kit.

Summary:

This kit contains resources for asynchronous lessons. There is a teacher guide to walk you through the challenges as well as learner handouts for 3 different challenges designed to take no more than 30 minutes. All challenges can be done offline.

In these activities, learners will explore how binary code can be used to represent images while exploring reflections and transformations. They will be invited to share their binary code with their peers and work from each other's codes. They will start with binary code for simple two-toned images and then work their way into coding four-toned images with possible extensions into how all colours are coded in binary for computers to use.

Kit Overview:

- Curriculum links
- Activity timeline suggestions
- Teacher's guide to challenges
- Learner challenge handouts

Required Technology and Materials:

For teachers:

- Computer with internet connection and ability to send learners a document

For learners:

- Computer or tablet with internet connection to receive challenge documents
- Ability to share and trade work with classmates
- Paper and pen/pencils
- Ability to print helpful but not necessary

Timeline:

Challenge 1: Representing and transmitting images using binary codes	30 minutes
Challenge 2: Binary coded shapes with four shades	30 minutes
Challenge 3: Binary coded shape transformations	30 minutes

Curriculum links:

[ADST](#)

Ideating

- Generate potential ideas and add to others' ideas
- Choose an idea to pursue

Testing

- Make changes and test again, repeating until satisfied with the product

Applied Skills

- Identify the skills required for a task and develop those skills as needed

Computational Thinking

- Simple algorithms that reflect computational thinking
- Visual representations of problems and data
- Binary number system to represent data

[Mathematics](#)

Reasoning and Analyzing

- Estimate reasonably
- Use tools or technology to explore and create patterns and relationships, and test conjectures
- Model mathematics in contextualized experiences

Understanding and solving

- Visualize to explore mathematical concepts
- Develop, demonstrate, and apply mathematical understanding through play, inquiry, and problem solving
- Apply multiple strategies to solve problems in both abstract and contextualized situations

Communicating and representing

- Communicate mathematical thinking in many ways
- Represent mathematical ideas in concrete, pictorial, and symbolic forms
- Explain and justify mathematical ideas and decisions

Connecting and reflecting

- Reflect on mathematical thinking
- Connect mathematical concepts to each other and to others areas of personal interests
- Use mathematical arguments to support personal choices

Curricular Content

- Line symmetry (grade 4)
- Single transformations (grade 5)
- Combinations of transformations (grade 6)

Core Competencies

- [Communicating](#)
 - Answering the reflection questions to explain their thinking
 - Communicating their ideas and explaining their code with their family, teacher, and peers
- [Creative thinking](#)
 - Identifying relationships between the binary codes, the area, and the perimeter to alter the shapes they have to fit new parameters
- [Critical and reflective thinking](#)
 - Analyzing their binary code given
- [Personal awareness and responsibility](#)
 - Persevering through difficult tasks and working towards finding a solution
 - If assigned as a self-paced lesson, managing your time appropriately and creating a plan for finishing the assignment
 - Developing and utilizing strategies to manage frustrating situations when coding
 - Seeking feedback