Computational Thinking

- The thought processes involved in formulating a problem and expressing it in such a way that another human or a computer could carry it out
- Involves:
  - **Decomposition** - breaking a big problem into a smaller one
  - **Abstraction** - remove parts of the problem that are unnecessary
  - **Pattern Recognition** - analyze & look for a repeating sequence
  - **Algorithm Design** - step-by-step instructions on how to do something

[Diagram of Computational Thinking]

https://www.computationalthinkers.com/product/computationalthinking/#prettyPhoto/0/
What is Scratch?

• Scratch is a graphical programming language, developed by the Lifelong Kindergarten group at the Massachusetts Institute of Technology (MIT).

• Drag and combine code blocks to make a range of programs, including animations, stories, musical instruments and games.

https://codeclubprojects.org/en-GB/resources/scratch-intro/
Open Scratch
• online https://scratch.mit.edu/
• or desktop
Start a new project
Move Forward
• Drag a Move Clock onto the code area
• Enter ‘50’ steps
• Click on the Move Block
• Watch the cat
Move Backwards
- Enter ‘-50’ steps
- Click on the Move Block
- Watch the cat
Turn
- Drag the Move block off the coding area
- Drag a Turn block into the coding area
- Enter 90 degrees
- Click on the Turn block
- Watch the cat
Drawing with a Pen

• Drag a <pen down>
• Drag a <set pen color to>
  • Change the number to 100
• Drag a <turn degrees>
• Draw a square
Draw a square with a loop

Make squares of different sizes

Draw a triangle, a hexagon, a pentagon....
Hacking Scratch

1. Open the file “Sprite chases the mouse” in Scratch and analyze each component.

2. Start changing the parameters (numbers) in the code to explore the effects in the program.

3. “Hack” the code to create your own version of the game.
Connecting with the Program of Studies

Open the Mathematics Program of Studies.

Where are the connections to coding and computational thinking?

Why is this way of thinking important in the mathematics classroom?
Why is Computational Thinking Important?

**Problem Solving**
- Develop problem solving strategies
- *How would you?* or *How could you?*
- Foster multiple creative and innovative solutions

**Reasoning**
- Thinking logically
- Make and test generalizations from patterns
- Use logical process to analyze and solve a problem

**Visualization**
- Thinking in pictures and images
- Pattern recognition
- Perceive, transform and recreate

**Number Sense**
- Order and number representation
- Relationships