### Lesson 1: Graph Paper Programming (unplugged)

**Objectives**
- Understand the difficulty of translating real problems into programs.
- Learn that ideas may feel clear and yet still be misinterpreted by a computer.
- Practice communicating ideas through codes and symbols.

**Themes**
- Algorithms

**Practices**
- Collaboration, Problem Solving

**Standards**
- ISTE: 1.b, 1.c, 2.d, 4.b, 4.d
- CSTA: CPP.L1:3-04, CPP.L1:6-05, CT.L1:3-03, CT.L1:6-01, CT.L1:6-02, CT.L2-07
- NGSS: K-2-PS3-2, 3-5-ETS1-2
- CC Mathematical Practices: 1, 2, 3, 6, 7, 8
- CC Math Standards: CC Math Standards: 2.G.2
- CC ELA: SL.1.1, SL.1.2, L.1.6
  - SL.2.1, SL.2.2, L.2.6
  - SL.3.1, SL.3.3, L.3.6

### Lesson 2: Real Life Algorithms – Paper Airplanes (unplugged)

**Objectives**
- Name various activities that make up their day.
- Decompose large activities into a series of smaller events.
- Arrange sequential events into their logical order.

**Themes**
- Algorithms

**Practices**
- Collaboration, Problem Solving

**Standards**
- ISTE: 1.c, 1.c, 2.b, 2.d, 4.b, 6.c
- CT.L2-06CC
- NGSS: K-2-PS3-2, 3-5-ETS1-2
- Mathematical Practices: 1, 2, 3, 6, 7, 8
- CC Math Standards: 1.G.1
  - 2.G.3
  - 3.G.2
- CC ELA: SL.1.1, SL.1.2, L.1.6
  - SL.2.1, SL.2.2, L.2.6
  - SL.3.1, SL.3.3, L.3.6

### Lesson 3: Maze - Sequence

**Objectives**
- Express movement as a series of commands
- Order movement commands as sequential steps in a program.
- Represent an algorithm as a computer program
- Count the number of times an action should be executed and represent
it as instructions in a program.
- Recall and apply the rules of pair programming.
- Use pair programming to complete collaborative tasks with or without a computer
- Identify situations when the rules of pair programming are not followed.

<table>
<thead>
<tr>
<th>Themes</th>
<th>Computing Practice and Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practices</td>
<td>Problem Solving</td>
</tr>
</tbody>
</table>
| Standards       | ISTE: 1.a, 1.c, 4.b, 6.a, 6.c, 6.d
                  | CSTA: CT.L1:3-01, CL.L1:3-02, CPP.L1:6-05, CPP.L1:6-06, CT.L2-01, CT.L2-06, CT.L2-08, CT.L2-12
                  | NGSS: K-2-PS3-2, 3-5-ETS1-2
                  | CC Mathematical Practices: 1, 2, 5, 6, 7, 8
                  | CC Math Standards: 1.OA.A.1
                  | 2.OA.A.1
                  | 3.OA.3
                  | CC ELA: SL.1.1, L.1.6
                  | SL.2.1, L.2.6
                  | SL.3.1, L.3.6

### Lesson 4: Artist - Sequence

| Objectives | Create a program to complete an image using sequential steps.
- Select an argument for a given command.
- Differentiate between defining and non-defining attributes of triangles, squares, and rectangles.
- Draw triangles, squares, and rectangles to reflect defining attributes.
- Explain the difference between squares and rectangles and support it with evidence consisting of the commands used to draw the different shapes.
- Compare and contrast squares and rectangles by their number of sides and side lengths.
- Compose two-dimensional shapes (rectangles, squares, trapezoids, triangles) to create a composite shape, such as two squares to compose a rectangle and two rectangles to compose a square.
- Compose new shapes from composite shapes.
- Draw partitions into a rectangle and describe the partitions using the words halves, fourths, quarters, half of, fourth of, and quarter of.
- Describe a whole rectangle as two halves or four quarters.
- Explain that decomposing into more equal shares creates smaller shares. |

<table>
<thead>
<tr>
<th>Themes</th>
<th>Computing Practice and Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practices</td>
<td>Creativity, Computing Practice and Programming</td>
</tr>
</tbody>
</table>
| Standards       | ISTE: 1.a, 1.c, 4.b, 6.a, 6.c, 6.d
                  | CSTA: CT.L1:3-01, CL.L1:3-02, CPP.L1:6-05, CPP.L1:6-06, CT.L2-01, CT.L2-06, CT.L2-08, CT.L2-12
                  | NGSS: K-2-PS3-2, 3-5-ETS1-2
                  | CC Mathematical Practices: 1, 2, 4, 5, 6, 7, 8
                  | CC Math Standards: 1.OA.A.1, 1.G.A.1, 1.G.A.2
                  | 2.OA.A.1, 2.G.A.1
                  | 3.OA.3, 3.G.A.2
                  | CC ELA: SL.1.1, L.1.6
                  | SL.2.1, L.2.6
**Lesson 5: Getting Loopy (unplugged)**

**Objectives**
- Repeat actions initiated by the instructor.
- Translate a picture program into a live-action dance.
- Convert a series of multiple actions into a single loop.

**Themes** Algorithms

**Practices** Collaboration, Problem Solving

**Standards**
- ISTE: 1.c, 2.d, 4.b, 6.a
- NGSS: K-2-PS3-2, 3-5-ETS1-2
- Mathematical Practices: 1, 2, 4, 6, 7, 8
- CC Math Standards: 1.MD.4
- CC ELA: SL.1.1, SL.1.2, L.1.6
  - SL.2.1, SL.2.2, L.2.6
  - SL.3.1, SL.3.3, L.3.6

**Lesson 6: Maze - Loops**

**Objectives**
- Identify the benefits of using a loop structure instead of manual repetition.
- Create a program for a given task which loops a single command.
- Break down a long sequence of instructions into the smallest repeatable sequence possible.
- Create a program for a given task which loops a sequence of commands.
- Employ a combination of sequential and looped commands to reach the end of a maze.

**Themes** Computing Practice and Programming

**Practices** Problem solving

**Standards**
- ISTE: 1.a, 1.c, 4.b, 6.a, 6.c, 6.d
- CSTA: CL.L1:3-02, CT.L1:3-01, CPP.L1:6-05, CPP.L1:6-06, CT.L2-01, CT.L2-06, CT.L2-08, CT.L2-12, CT.L3A-03
- NGSS: K-2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2
- CC Mathematical Practices: 1, 2, 4, 5, 6, 7, 8
- CC Math Standards: 1.OA.A.1
  - 2.OA.A.1
  - 3.OA.A.3
- CC ELA: SL.1.1, L.1.6
  - SL.2.1, L.2.6
  - SL.3.1, L.3.6

**Lesson 7: Artist - Loops**

**Objectives**
- Count the number of times an action should be repeated and represent it as a loop.
- Decompose a shape into its smallest repeatable sequence.
- Create a program that draws complex shapes by repeating simple sequences.

**Themes** Computing Practice and Programming

**Practices** Problem solving

**Standards**
- ISTE: 1.a, 1.b, 1.c, 4.b, 6.a, 6.c, 6.d
### Lesson 8: Bee - Loops

**Objectives**
- Write a program for a given task which loops a single command.
- Identify when a loop can be used to simplify a repetitive action.
- Employ a combination of sequential and looped commands to move and perform actions.

**Themes**
Computing Practice and Programming

**Practices**
Problem solving

**Standards**
ISTE: 1.a, 1.c, 4.b, 6.a, 6.c, 6.d
CSTA: CL.L1:3-02, CT.L1:3-01, CPP.L1:6-05, CPP.L1:6-06, CT.L2-01, CT.L2-06, CT.L2-08, CT.L2-12, CT.L3A-03
NGSS: K-2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2
Mathematical Practices: 1, 2, 4, 5, 6, 7, 8
CC Math Standards: 1.OA.1, 2.OA.1, 2.G.A.1
CC ELA: SL.1.1, L.1.6
SL.2.1, L.2.6
SL.3.1, L.3.6

---

### Lesson 9: Relay Programming (unplugged)

**Objectives**
- Practice communicating ideas through codes and symbols
- Use teamwork to complete a task
- Verify the work of their teammates to ensure a successful outcome

**Themes**
Algorithms

**Practices**
Collaboration, Problem Solving

**Standards**
ISTE: 1.a, 1.c, 2.d, 4.b, 4.d, 6.a
CSTA: CT.L1:3-01, CT.L1:3-03, CT.L1:6-01, CT.L1:6-02, CT.L1:6-05, CT.L2-01, CT.L2-03, CT.L2-06, CT.L2-07, CT.L2-08, CT.L2-12, CPP.L1:3-04, CPP.L1:6-05
NGSS: K-2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2
Mathematical Practices: 1, 2, 3, 6, 7, 8
CC Math Standards: 2.G.2
CC ELA: SL.1.1, SL.1.2, SL.2.1, L.2.6
SL.3.1, L.3.3, L.3.6
## Lesson 10: Bee Debugging

**Objectives**
- Predict where a program will fail.
- Modify an existing program to solve errors.
- Identify an algorithm that is unsuccessful when the steps are out of order.
- Reflect on the debugging process in an age-appropriate way.

**Themes**
- Algorithms, Computing Practice and Programming

**Practices**
- Persistence, Problem Solving

**Standards**
- ISTE: 1.a, 1.c, 4.b, 4.c, 4.d, 6.a, 6.c, 6.d
- CSTA: CL.L1:3-02, CT.L1:3-01, CT.L1:6-01, CT.L2-01, CT.L2-06, CT.L2-08, CT.L2-12 CPP.L1:6-05, CPP.L1:6-06, CT.L3A-03
- NGSS: K- 2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2
- Mathematical Practices: 1, 2, 4, 5, 6, 7, 8
- CC Math Standards: 1.OA.1, 2.OA.1, 3.OA.3
- CC ELA: SL.1.1, L.1.6, SL.2.1, L.2.6, SL.3.1, L.3.6

## Lesson 11: Artist - Debugging

**Objectives**
- Predict where a program will fail.
- Modify an existing program to solve errors.
- Identify an algorithm that is unsuccessful when the steps are out of order.
- Reflect on the debugging process in an age-appropriate way.
- Something about calculating angles / measuring distance.

**Themes**
- Algorithms, Computing Practice and Programming

**Practices**
- Persistence, Problem Solving

**Standards**
- ISTE: 1.a, 1.c, 4.b, 4.d, 6.a, 6.c, 6.d
- CSTA: CL.L1:3-02, CT.L1:3-01, CT.L1:6-01, CT.L2-01, CT.L2-06, CT.L2-07, CT.L2-08, CT.L2-12 CPP.L1:6-05, CPP.L1:6-06, CT.L3A-03
- NGSS: K- 2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2
- Mathematical Practices: 1, 2, 4, 5, 6, 7, 8
- CC ELA: SL.1.1, L.1.6, SL.2.1, L.2.6, SL.3.1, L.3.6

## Lesson 12: Conditionals (unplugged)

**Objectives**
- Define circumstances when certain parts of programs should run and when they shouldn’t.
- Determine whether a conditional is met based on criteria.
- Traverse a program and predict the outcome, given a set of input.

**Themes**
- Algorithms

**Practices**
- Problem solving

**Standards**
- ISTE: 1.a, 1.c, 2.d, 4.b, 6.a
- CSTA: CT.L1:3-03, CT.L1:6-01, CT.L1:6-02, CT.L1:6-05, CPP.L1.3-04, CPP.L1:6-
## Lesson 13: Bee - Conditionals

**Objectives**
- Compare binary values.
- Translate spoken language conditional statements into a program.
- Identify when a conditional can be used to deal with unknown values.
- Execute an algorithm with a conditional statement.
- Solve puzzles using a combination of looped sequences and conditionals.

**Themes**
Computing Practice and Programming

**Practices**
Problem solving

**Standards**
ISTE: 1.a, 1.c, 4.b, 4.d, 6.a, 6.c, 6.d  
CSTA: CT.L1:3-02, CT.L1:3-03, CPP.L1:6-05, CPP.L1:6-06, CT.L1:6-01, CT.L2-01, CT.L2-06, CT.L2-07, CT.L2-08, CT.L2-012, CT.L2-14, CT.L3A-03  
NGSS: K-2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2  
Mathematical Practices: 1, 2, 4, 5, 6, 7, 8  
CC Math Standards: 1.OA.1, 2.OA.1, 2.G.2, 2.MD.5, 2.NBT.A.4, 3.OA.3  
CC ELA: SL.1.1, L.1.6, SL.2.1, L.2.6, SL.3.1, L.3.6

## Lesson 14: Binary Bracelets (unplugged)

**Objectives**
- Encode letters into binary.
- Decode binary back to letters.
- Relate the idea of storing initials on a bracelet to the idea of storing information in a computer.

**Themes**
Data

**Practices**
Creativity

**Standards**
ISTE: 1.a, 1.c, 2.d, 4.b, 6.a, 6.d  
CSTA: CT.L1:3-03, CT.L1:6-03, CT.L1:6-05, CT.L2-07, CT.L2-08  
NGSS: K-2-PS3-2, K-2-ETS1-1  
Mathematical Practices: 1, 2, 4, 6, 7, 8  
CC ELA: SL.1.1, SL.1.2, L.1.6, SL.2.1, SL.2.2, L.2.6, SL.3.1, SL.3.3, L.3.6

## Lesson 15: The Big Event (unplugged)

**Objectives**
- Repeat commands given by an instructor.
- Recognize actions of the teacher as signals to initiate commands.
- Practice differentiating pre-defined actions and event-driven ones.

**Themes**
Algorithms

**Practices**
Creativity, Collaboration
## K–5 Curriculum Course 2

### Framework

<table>
<thead>
<tr>
<th>Standards</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTE: 1.a, 4.b, 6.a, 6.c, 6.d</td>
<td></td>
</tr>
<tr>
<td>NGSS: K-2-ETS1-1</td>
<td></td>
</tr>
<tr>
<td>Mathematical Practices: 1, 2, 6, 7, 8</td>
<td></td>
</tr>
<tr>
<td>CC ELA: SL.1.1, SL.1.2, L.1.6</td>
<td></td>
</tr>
<tr>
<td>SL.2.1, SL.2.2, L.2.6</td>
<td></td>
</tr>
<tr>
<td>SL.3.1, SL.3.3, L.3.6</td>
<td></td>
</tr>
</tbody>
</table>

### Lesson 16: Flappy

**Objectives**
- Match blocks with the appropriate event handler.
- Create a game using event handlers.
- Share a creative artifact with other students.

**Themes**
Computing Practice and Programming

**Practices**
Persistence, Problem Solving

<table>
<thead>
<tr>
<th>Standards</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTE: 1.a, 1.b, 1.c, 4.b, 6.a, 6.c, 6.d</td>
<td></td>
</tr>
<tr>
<td>CSTA: CL.L1:3-02, CL.L1:6-01, CPP.L1:6-05, CPP.L1:6-06, CT.1:3-02, CT.L1:6-01, CT.L2-01, CT.L2-06, CT.L2-07, CT.L2-08, CT.L2-12</td>
<td></td>
</tr>
<tr>
<td>NGSS: K-2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2</td>
<td></td>
</tr>
<tr>
<td>CC Mathematical Practices: 1, 2, 5, 6, 7, 8</td>
<td></td>
</tr>
<tr>
<td>CC ELA: SL.1.1, SL.1.2, L.1.6</td>
<td></td>
</tr>
<tr>
<td>SL.2.1, SL.2.2, L.2.6</td>
<td></td>
</tr>
<tr>
<td>SL.3.1, SL.3.3, L.3.6</td>
<td></td>
</tr>
</tbody>
</table>

### Lesson 17: Play Lab – Create a Story

**Objectives**
- Identify actions that correlate to input events.
- Create an animated, interactive story using sequence, loops, and event-handlers.
- Share a creative artifact with other students.

**Themes**
Computing Practice and Programming

**Practices**
Creativity, Problem Solving, Collaboration

<table>
<thead>
<tr>
<th>Standards</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTE: 1.a, 1.b, 1.c, 4.b, 6.a, 6.c, 6.d</td>
<td></td>
</tr>
<tr>
<td>CSTA: CT.L1:3-01, CL.L1:3-02, CL.L1:6-01, CPP.L1:3-03, CPP.L1:6-03, CPP.L1:6-05, CPP.L1:6-06, CT.L2-01, CT.L2-06, CT.L2-07, CT.L2-08, CT.L2-12</td>
<td></td>
</tr>
<tr>
<td>NGSS: K-2-PS3-2, K-2-ETS1-1, 3-5-ETS1-2</td>
<td></td>
</tr>
<tr>
<td>CC Mathematical Practices: 1, 2, 5, 6, 7, 8</td>
<td></td>
</tr>
<tr>
<td>CC Math Standards: 1.OA.A.1</td>
<td></td>
</tr>
<tr>
<td>2.OA.A.1, 2.MD.5</td>
<td></td>
</tr>
<tr>
<td>CC ELA: SL.1.1, SL.1.5, L.1.6, W.1.6</td>
<td></td>
</tr>
<tr>
<td>SL.2.1, SL.2.5, L.2.6, W.2.3</td>
<td></td>
</tr>
<tr>
<td>SL.3.1, SL.3.6, W.3.3. W3.6</td>
<td></td>
</tr>
</tbody>
</table>

### Lesson 18: Your Digital Footprint (unplugged)

**Objectives**
- Understand that being safe when they visit websites is similar to staying safe in real life.
- Learn to recognize websites that are alright for them to visit.
- Recognize if they should ask an adult they trust before they visit a particular website.
- Explore what information is appropriate to be put online.

**Themes**
Community Global and Ethical Impacts

**Practices**
Communicating
| Standards | ISTE: 5.a, 5.b, 6.a  
CSTA: CI.L1:3-01, CPP.L2-06  
CC ELA: SL.1.1, SL.1.2, L.1.6  
SL.2.1, SL.2.2, L.2.6  
SL.3.1, SL.3.3, L.3.6 |