Title: "Math in Motion: Programming Geometric Patterns"

Objective: Students will use mathematical concepts to create and modify simple programs that generate geometric patterns, incorporating control structures like loops and conditionals.

Duration: 60 minutes

Materials:

- Computers with a simple block-based programming environment (e.g., Scratch)
- Graph paper for planning designs
- Handouts with basic programming concepts and geometric formulas

Lesson Plan:

1. Introduction (10 minutes):
    - Briefly review basic geometric shapes and their properties.
    - Introduce the concept of using programming to create geometric patterns.
    - Demonstrate a simple program that draws a square using a loop.
2. Basic Programming Concepts (15 minutes):
    - Explain key programming concepts:
        - Variables (to store numbers)
        - Loops (to repeat actions)
        - Conditionals (to make decisions)
    - Show how these concepts relate to mathematical operations and logic.
3. Guided Programming Activity (20 minutes):
    - Walk students through creating a program that draws a regular polygon.
    - Use a loop to repeat the drawing process.
    - Incorporate a variable for the number of sides.
    - Use the formula for the exterior angle of a regular polygon:
      Exterior Angle = $\frac{360^\circ}{n}$, where n is the number of sides

    - Have students experiment with different values for n to create various polygons.

4. Creative Challenge (10 minutes):
    - Challenge students to modify the program to create a spiral pattern.
    - Introduce the concept of incrementing a variable within a loop.
    - Encourage students to use mathematical thinking to determine how to change the side length or angle in each iteration.
5. Problem-Solving Extension (10 minutes):
    - Present a problem: "Create a program that draws a star pattern."
    - Guide students to think about:
        - How many points should the star have?
        - What angles are needed to create the star shape?
        - How can they use loops and conditionals to alternate between inward and outward points?
6. Reflection and Sharing (5 minutes):
    - Have students share their creations and explain the math behind their designs.
    - Discuss how mathematical concepts (geometry, variables, formulas) were used in programming.

Math Concepts Covered:

- Geometry (polygons, angles, spirals)
- Variables and equations
- Sequences and patterns
- Coordinate plane (for positioning shapes)

Programming Concepts Covered:

- Variables
- Loops (for and while loops)
- Conditionals (if-then statements)

- Basic algorithm design

Extension Activities:

1. Challenge advanced students to create more complex patterns using nested loops or fractals.
2. Assign a project where students create an interactive math art piece, allowing users to input values that affect the pattern.

Assessment:

- Evaluate students' programs for correct use of mathematical concepts and programming structures.
- Assess students' ability to explain the mathematical principles behind their designs.
- Consider a follow-up activity where students analyze and modify each other's code.

Sample Code Structure (in pseudocode):

```
Variable sides = 5  // Number of sides in the polygonVariable length = 50 // Length of each sideRepeat sides times:    Move forward length    Turn right (360 / sides) degrees// For creating a spiralRepeat 50 times:    Move forward length    Turn right (144) degrees  // For a five-pointed star pattern length = length + 2  // Increase side length each iteration
```

This lesson plan integrates mathematical skills with basic programming concepts, aligning with the NYSED CSDF standard while reinforcing 7th-grade math concepts. It encourages students to think critically about how math is applied in programming and how it can be used for creative expression and problem-solving.