

**CODE/MOE/UOIT Makerspaces Project--Lesson Planning Template**

**School Board: Rainy River District School Board**

**Grade(s): 3-4**

**Subject(s):Math—Geometry, Patterning and Algebra**

|  |  |
| --- | --- |
| **BIG IDEAS:**  **Basic coding involves all the skills of problem solving: reasoning, communicating, reflecting and metacognition.**  **Curriculum Expectations:**  **OVERALL: Gr. 3** use a reference tool to identify right angles and to compare angles with a right angle;  **Gr**. **3** classify two-dimensional shapes by geometric properties (number of sides and angles);  **Gr. 4** classify two-dimensional shapes by geometric properties (number of sides, angles, and symmetry);  **SPECIFIC:**  **Gr. 3** -use a reference tool (e.g., paper corner, pattern block, carpenter’s square) to identify right angles and to describe angles as greater than, equal to, or less than a right angle  -identify and compare various polygons (i.e., triangles, quadrilaterals, pentagons, hexagons, heptagons, octagons) and sort them by their geometric properties (i.e., number of sides; side lengths; number of interior angles; number of right angles)  **Gr. 4** compare, using a variety of tools (e.g., geoboard, patterns blocks, dot paper, Scratch program), two-dimensional shapes that have the same perimeter or the same area (Sample problem: Draw, using grid paper, as many different rectangles with a perimeter of 10 units as you can make on a geoboard, or in the Scratch computer program at scratch.mit.edu).  identify benchmark angles (i.e., straight angle, right angle, half a right angle), using a reference tool (e.g., paper and fasteners, pattern blocks, straws), and compare other angles to these benchmarks (e.g.,“The angle the door makes with the wall is smaller than a right angle but greater than half a right angle.”) (Sample problem: Use paper folding to create benchmarks for a straight angle, a right angle, and half a right angle, and use these benchmarks to describe angles found in pattern blocks.);  – relate the names of the benchmark angles to their measures in degrees (e.g., a right angle is 90º);  – identify and describe the general location of an object using a grid system (e.g. describe the location of co-ordinates of a 2 dimensional shape using the grid on a Scratch screen)  -make predictions related to repeating geometric and numeric patterns (Sample problem: Create a block program in Scratch using the repeat command that will draw 3, 4, 5, 6, or more sided shapes).  -determine, through investigation, the relationship between the number of sides of regular 2 dimensional figures and the number of times to use the repeat command in Scratch to draw the figure) | |
| **Learning Goals:**  “We are learning to…”  **Use the coding program Scratch to draw regular 2 dimensional figures using as few programming commands as possible.** | **Success Criteria:**  “We will be successful when we…”  **Code the most effective block programming to draw any 2 dimensional regular shape.**  *When I code, I have to:*   1. *Record my coding so I have a record of what I did* 2. *Use my critical thinking skills to find out why my coding did not work* 3. *Debug any parts of my coding that did not work the way I expected it to* 4. *Plan my next coding project incorporating new coding commands I will learn* |
| **Lesson Overview:**  Teacher will model some of the basic Scratch command to students such as pen up, pen down, right, left, forward, back, repeat, and how to shrink a sprite. Then we will create a code to make a square together. Next students will code drawing another 2 dimensional shape. | |
| **Materials and Technology:**  Protractors  Square corners  Chromebooks or Computers with the Scratch program  A notebook to record coding | |
| **Student Accommodations/Modifications:** | **Lesson will be differentiated by:**   * **Content, specifically:** * **Process, specifically:**   **X Product, specifically:** some students may only figure out how to program the shapes not using the repeat command   * **Environment, specifically:** |
| **MINDS ON: Getting Started** | |
| During this phase, the teacher may:  • activate students’ prior knowledge;  • engage students by posing thought-provoking questions;  • gather diagnostic and/or formative assessment data through observation and questioning;  • discuss and clarify the task(s). | During this phase, students may:  • participate in discussions;  • propose strategies;  • question the teacher and their classmates;  • make connections to and reflect on prior learning. |
| **Describe how you will introduce the learning activity to your students.**  ***Minds On--Before/Activation of Prior Knowledge (5-7 minutes)***–  1 Students code/program the teacher to move from the teacher’s desk to the window (or other point in the room) according to the directions the class gives them. (e.g. turn right 90 degrees, go forward 15 steps, turn 90 degrees right, go forward 5 steps etc.)  **What key questions will you ask?**  Students are asked to ***Think-Pair-Share*** with their elbow partner, about how easy or difficult it was to get the teacher to the window.  **How will you gather diagnostic or formative data about the students’ current levels of understanding?**  Have pairs share their discussions with the whole group.  As a big group teacher records student responses.  Teacher highlights the “I-Wonder” question *How can I most effectively communicate movement of a person or object in space or in 2 dimensions?*  **How will students be grouped? How will materials be distributed?**  Students will work on their own chromebook/computer if in a one to one school, if not, then partners will discuss and share the device, or the activities can be done in small groups. | |
| **ACTION: Working on it** | |
| During this phase, the teacher may:  • ask probing questions;  • clarify misconceptions, as needed, by redirecting students through questioning;  • answer students’ questions (but avoid providing a solution to the problem);  • observe and assess;  • encourage students to represent their thinking concretely and/or pictorially;  • encourage students to clarify ideas and to pose questions to other students. | During this phase, students may:  • represent their thinking (using numbers, pictures, words, manipulatives, actions, etc.);  • participate actively in whole group, small group, or independent settings;  • explain their thinking to the teacher and their classmates;  • explore and develop strategies and concepts. |
| **Describe the task(s) in which your students will be engaged.**  Teacher will model some of the basic Scratch command to students such as pen up, pen down, right, left, forward, back, repeat, and how to shrink a sprite. Then we will create a code to make a square together.  **Lesson Problem:**  *Students will code a sprite in the program Scratch to draw a rectangle, and then choose another 2D figure such as a pentagon, or hexagon.*  Students will break off and work on their own Chromebooks. Students will be presented with a coding book to record their coding and pictures of what they want to code Students will be given 5 minutes to write in their books about what worked well in their coding and what did not work, and why it did not work. Teacher/E.A. will walk around and note students successes and attempts, or assist students who are off task, or do not know where to start.  **What misconceptions or difficulties do you think they might experience?**  -forgetting to put the pen down to see the results of their coding  -choosing a sprite that is difficult to determine which way the head is pointing (e.g. a ball)  -using the repeat command too many times so the sprite is drawing over itself (not efficient programming  -losing their sprite if they have their sprite go too many steps and the sprite disappears off the screen  **How will they demonstrate their understanding of the concept?**  Students will code several shapes of their choice (e.g. a rectangle, pentagon, hexagon, etc.)  **How will you gather your assessment data (e.g., checklist, anecdotal records)?**  Checklist of shapes attempted successfully.  Anecdotal notes on the difficulties students had.  Look at the student coding books where they write their coding steps and draw their pictures of their results.  Students also had a writing assignment that afternoon to describe their successes and difficulties in coding.  **What extension activities will you provide?**  Show students how to use the repeat command  Challenge will be to code a triangle or a circular looking shape.  Move your 2D figure to a new location on the grid.  Plan what you would like to code a sprite to do next time you go in Scratch. | |
| **CONSOLIDATION: Reflecting and Connecting** | |
| During this phase, the teacher may:  • bring students back together to share and analyse strategies;  • encourage students to explain a variety of learning strategies;  • ask students to defend their procedures and justify their answers;  • clarify misunderstandings;  • relate strategies and solutions to similar types of problems in order to help students generalize concepts;  • summarize the discussion and emphasize key points or concepts. | During this phase, students may:  • share their findings;  • use a variety of concrete, pictorial, and numerical representations to demonstrate their understandings;  • justify and explain their thinking;  • reflect on their learning. |
| **How will you select the individual students or groups of students who are to share their work with the class (i.e., to demonstrate a variety of strategies, to show different types of representations, to illustrate a key concept)?**  Students will come back together into a big group. Using student work samples, teacher will ask some students to explain their choice of a code to get their sprite to draw a square, and another 2D shape. Teacher will compose a list of successful strategies on the Smart Board as the students explains them from their own work.  **What key questions will you ask during the debriefing?**  ***Effective Questions for teacher to ask students:***   1. Can you think of a way to debug your coding? 2. How would a coder represent their thinking? 3. Is there another more efficient way to code this problem? 4. Which coding do you find more effective, coding with the Ozobots, or with Scratch? Why was it more effective for you? | |