Lesson Planning Template **Teacher: DeSIRE**

📆Date:	Topic: Touch Control	Time Required: 90 minutes

🎯**Learning Target/Objectives:**

- **I can** successfully configure the Arduino IDE to compile and upload sensor-based code to a robotic controller .
- **I can** explain the program logic used to correlate the number of sensor touches with specific robotic transport destinations .
- **I can** modify existing source code to change the default behavior of a robot's automated sorting process.

📜**Vocabulary:**

- **Touch Sensor**
- **Arduino/Genuino Uno**
- **COM Port**
- **Ino Program**
- **Program Logic**
- **Function Extension**

❓**Guiding Questions:**

- How does the robot distinguish between a single touch and multiple continuous touches?
- Why is it important to select the correct "Board" and "Port" in the software before trying to upload code ?
- In what way does the "(result+12)" mathematical parameter in the code relate to the physical movement of the arm?

📚**Lesson Design Details:**

- **Activity 1: The Code Navigator**
  - **Focus:** Students navigate the Arduino IDE to locate line 84. They must identify the runActionGroup command and explain what the current number inside the parentheses does .
- **Activity 2: Sorting Logic Challenge**
  - **Focus:** Students set up the physical map with color blocks. They must perform a "Validation Run" where they touch the sensor 1, 2, and 3 times and document if the blocks reach the correct "Position after transport" .
- **Activity 3: The "Middle Pick" Modification**
  - **Focus:** Students perform a "Live Edit." They modify the code from (result+12) to (result+13) to change which block the robot picks up first. They must re-compile and upload to verify the change.

Key Points (Vocabulary):

- **Touch Sensor:** An electronic sensor that detects and records physical contact to trigger a digital signal.
- **Arduino/Genuino Uno:** The specific micro-controller board used as the "brain" for processing the touch sensor data.
- **COM Port:** The communication channel on a computer used to upload data to the robotic hardware .
- **Ino Program:** The file format used for Arduino sketches (source code).
- **Program Logic:** The sequence of "if-then" style instructions that tell the robot how to respond to inputs.
- **Function Extension:** The process of adding or modifying code to change or expand a robot's capabilities.

Key Points of Instruction

- **Environment Synchronization:** Students must understand that for the hardware to work, the "Brain" (Arduino Uno) must be synced with the "Editor" (IDE) via the correct COM port .
- **Input-Process-Output (IPO):** Use this lesson to illustrate the IPO model. The Input is the human touch, the Process is the Arduino logic counting those touches, and the Output is the arm moving to a coordinate.
- **Code as a Map:** Explain that `runActionGroup` is a command that calls up a "map" of movements already stored in the robot's memory.
- **Precision and Safety:** Emphasize that during the upload process, moving the USB cable can cause a "Program upload failed" error.

Teacher's Cheat Sheet	
Feature	**Technical Data / Value**
Board Type	Arduino/Genuino Uno
Code File	Touch_control.ino
Upload Indicator	"Done uploading" prompt in the IDE
Right Side Grab	(result+12) in code

Middle Grab	(result+13) in code
Left Side Grab	(result+14) in code
Safety Warning	Do not move the USB cable during upload
Serial Connection	5V via USB cable

Category	Standard Organization	Standard/Benchmark Code and Description
Technology	ITEEA	**STEL-2R:** Follow step-by-step instructions to safely use systems and troubleshoot common problems
Computer Science	NCSOS	**HS-CS-03:** Illustrate the ways computing systems implement logic, input, and output through hardware components
Computer Science	NCSOS	**HS-AP-10:** Create procedures with parameters to organize code and make it easier to reuse
Digital Literacy	ISTE	**1.5.d:** Students understand how automation works and use algorithmic thinking to develop a sequence of steps
Engineering	ITEEA	**STEL-7Q:** Apply a broad range of making skills to follow a design process in the construction of a prototype 5.