



Translating Rock Paper Scissors

The following Scratch code determines who wins if the computer chooses ROCK. It uses **complex conditionals**. Please translate it into Arduino code. For the “say” block, you may call a say function, for example: `say("It's a tie!");`

SCRATCH	ARDUINO
	<pre> if (computer == player) { say("It's a tie"); } if (computer == ROCK && player == PAPER) { say("You win, Paper covers Rock"); } if (computer == ROCK && player == SCISSORS) { say("You lose, Rock breaks Scissors"); } </pre>

The following Scratch code also determines who wins if the computer chooses ROCK. It uses **nested conditionals**. Please translate it into Arduino code. For the “say” block, you may call a say function, for example: `say("It's a tie!");`

SCRATCH	ARDUINO
	<pre> if (computer == player) { say("It's a tie"); } else { if (computer == ROCK) { if (player == PAPER) { say ("You win, Paper covers Rock"); } else { say("You lose, Rock breaks Scissors"); } } } </pre>

Do you prefer complex conditionals or nested conditionals to write your code? Why?

ANSWERS WILL VARY.