Coding Microbits using Python — Reflections

Module 4: Making Decisions (Conditionals)

Computer programs are instructions telling the computer how to process input and deliver output. An important part of programming is telling the computer WHEN to perform a certain task. For this, we use something called ‘conditionals’. Conditionals get their name because a certain Condition or Rule has to be met. Conditionals are usually implemented using an ‘if (condition) then action statement. Students will be creating and making projects like coin toss, Magic 8 Ball, and/or dice toss with dots instead of numbers.

Module 4: Making Decisions (Conditionals)
List 3 decisions you have made today.

_____

_____

_____

Conditionals:

if (*condition*) then

    Action if true

else

    Action if condition is false

Red Light - Green Light conditionals:

if (_____) then

    _____

if (_____) then

    _____

if (_____) then

    _____

else

    _____

Conditions and Boolean expressions

All of the conditions in an if...then statement have to be an expression that can be evaluated as either True or False. These are called Boolean expressions when they are either True or False. These expressions usually use comparison operators to decide if it is True or False.

Comparison Operators		
Operator	Name	Example
==	equals	`x == y`
!=	not equal	`x != y`
>	greater than	`x > y`
<	less than	`x < y`
>=	greater than or equal to	`x >= y`
<=	less than or equal to	`x <= y`

Logical Operators		
Operator	Description	Example
and	Returns True if both statements are true	x < 5 and x < 10
or	Returns True if one of the statements are true	x < 5 or x < 4
not	Reverse the result, returns False if the result is true	not(x < 5 and x < 10)

04.2a Coin Toss Activity

Algorithm & Pseudocode:

04.2b Dice Roll Activity

Algorithm & Pseudocode:

04.3 Project: Board Game (done with a partner)

In this project you will plan, design, and create a board game. It should have clear rules on how to play. It should use conditionals on the microbit in a way that is central to the game. It should also use a maker elements as part of the design and construction. (Do a search for “DIY board games.)

Brainstorm Ideas _____

_____

_____

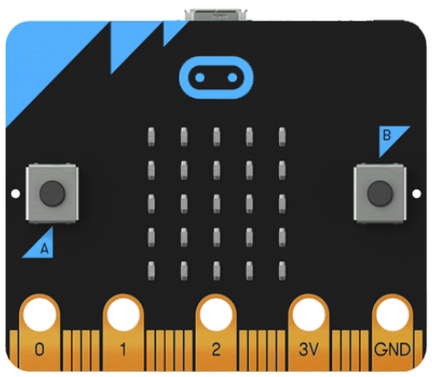
_____

Project: _____

Description: _____

_____

Microbit Project Sketch:



Game Board Sketches

Board Game Rules and Conditionals in Playing:

Board Game Algorithm & Pseudocode:

Materials Needed: _____

Coding Plan: _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Photos:

Notes & Reflections

How did you decide with your partner on your board game? _____

_____

_____

_____

What was something that was surprising to you in the creation of your board game?

_____

_____

_____

How well did your prototype work? _____

_____

Describe a difficult point in designing your game and how you resolved it:

_____

_____

_____

What feedback did you get from your beta testers?

_____

_____

What did you change to improve your game? _____

_____

Describe how you and your partner shared the work on the project.

_____

_____

_____

Assessment Rubric

Competency scores

Competency	4	3	2	1
Rules	All game rules are clear and complete.	A game rule is missing or not complete or not clear.	More than one game rule is missing or not complete or not clear.	Most of the game rules are missing or it is not clear what the rules are.
Game Board	Game board is: 1) Complete 2) Neat 3) Fits with the theme of the game 4) micro:bit is a central part of the game	Game board meets only 3 of the conditions listed for a score of 4.	Game board meets only 2 of the conditions listed for a score of 4.	Game board meets only 1 of the conditions listed for a score of 4.
Micro:bit Program	micro:bit program: 1) Uses the micro:bit in a way that is integral to the game 2) Uses conditionals correctly 3) Compiles and runs as intended 4) JavaScript includes comments in code	micro:bit program lacks 1 of the required elements.	micro:bit program lacks 2 of the required elements.	micro:bit program lacks 3 of the required elements.
Photo Documentation	Complete photo documentation that includes photos of game board and code and captions.	A photo is missing or of poor quality or a caption is missing.	Multiple photos and/or captions missing or of poor quality.	Most photos and/or captions missing or of poor quality.
Collaboration Reflection	Reflection piece includes: 1) Brainstorming ideas 2) Construction 3) Programming 4) Beta testing	Reflection piece lacks 1 of the required elements.	Reflection piece lacks 2 of the required elements.	Reflection piece lacks 3 of the required elements.

Comments or Photos:

Notes