

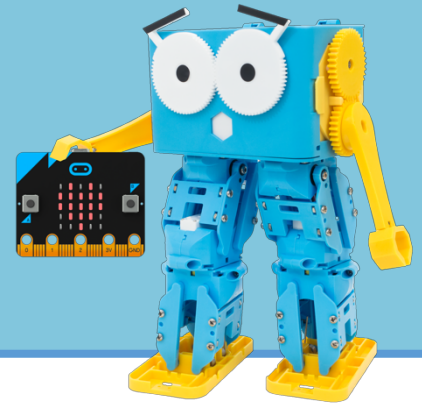
Lesson 1.23 – Hack Snakes & Ladders

Education Level: Second Level (Age 7-11)

Lesson Duration: 45 minutes

Prerequisite Knowledge: Lessons 1.1-1.14 & 1.19-1.20

Device Compatibility: Laptop, PC or Tablet



Lesson Overview

In this lesson, students will explore snakes and ladders with a twist where Marty is the counter and our Micro:Bit is the dice. Students will need to think about what a dice is and how it works before thinking about how they are going to play the games and what additional rules they want to add to the game.

Learning Objectives

- Consider the attributes of a dice and write a program to recreate one on the Micro:Bit
- Program Marty to act as our counter on the snakes and ladders board
- Explore additional rules and ideas to extend the game and make it their own

Key Vocabulary

- Micro:Bit
- Random number
- User Interaction
- Rules
- Event
- Computational Thinking

Resources & Equipment

- Marty the Robot
- Marty Workbook (Lesson 5)
- Laptops/Computers/Tablets
- Access to the Scratch 3 editor
- BBC Micro:Bit
- Snakes and ladders game board

Additional Reading

- Educator's Guide
- Introduction to Programming with Marty using Scratch
- BBC Micro:Bit Quick Start Guide for Teachers (<https://microbit.org/guide/quick/>)

Learning Plan & Activities

1. Ask students to describe how you play snakes and ladders
 - a. What do you need? How do you set it up? What are the rules? Who goes first? What happens when you take your first turn? And the rest of the turns? How do you know when the game ends?
2. Start going through the list of things that we need to play but we realise that we have forgotten a dice, is there anything that we could use to replace the dice?
 - a. What is a dice? How does it work? Can I get the same number every time? What device has a screen to display the number? What device has sensors in it so that I can shake it to *roll* a number?
3. Break students into small groups of around 2-3 per Marty and Micro:Bit, the team then has a series of programming tasks which they should test each time that they add something new
 - a. Firstly, they need to make a dice on the Micro:Bit starting off with displaying a random number and then ensuring that this random number appears when you shake the device
 - b. Next, we need to program Marty to respond to the number we have just rolled on our dice, he needs to walk forwards the correct number of steps

- c. What about when we land on a snake or a ladder? We will need to roll again but when we land on a snake, we will need to tell our program that we need to go backwards instead of forwards – are there any other gestures we could use like clicking buttons A and B at the same time to do this?
4. Students should challenge another student group to a game of snakes and ladders to fully test out their programs and see who can win

Additional Challenges

- What other rules could be added onto the game to make it easier or more difficult? For example, if you land on a certain coloured square then you have to answer a computing question correctly to stay there
- What would make it easier for you to win? Could you implement this into your dice secretly without your opponent finding out?
- Can you add in a winning dance for when Marty reaches the end of the game? Could you program him to do this movement without prompting him first? (For example, keeping track of what square you are on in your program?)

Curriculum Benchmarks

Curriculum for Excellence – Technologies Benchmark Guide

● = Fully Addresses Benchmark ○ = Partially Addresses Benchmark

Curriculum Organiser	Benchmark Covered	Lesson 1.23
Digital Literacy	TCH 0-01a	●
Technological Developments in Society & Business	TCH 0-05a	●
Craft, Design, Engineering and Graphics	TCH 0-11a	●
	TCH 1-12a	●
Computing Science	TCH 0-13a	●
	TCH 1-13a	●
	TCH 2-13a	●
	TCH 3-13a	●
	TCH 3-13b	●
	TCH 4-13a	○
	TCH 0-14a	●
	TCH 0-14b	●
	TCH 1-14a	●
	TCH 1-14b	●
	TCH 2-14a	●
	TCH 3-14a	●
	TCH 0-15a	●
	TCH 1-15a	●
	TCH 2-15a	●
	TCH 3-15a	○
	TCH 4-15a	○

National Curriculum – Computing, Design & Technology

● = Fully Addresses Benchmark ○ = Partially Addresses Benchmark

Curriculum Organiser	Benchmark Covered	Lesson 1.23
Computing	1-a	●
	1-b	●
	1-c	●
	1-e	●
	2-a	●
	2-b	●
	2-c	●
	2-f	●
	3-a	●
	3-b	○
	3-d	○
	4-a	○
	4-b	○
	Design & Technology	1.1-b
1.2-a		○
2.3-a		●
2.3-b		●

Australian F-10 Curriculum – Digital Technologies, Design & Technologies

● = Fully Addresses Benchmark ○ = Partially Addresses Benchmark

Curriculum Organiser	Benchmark Covered	Lesson 1.23
Digital Technologies	ACTDIK001	●
	ACTDIK002	●
	ACTDIP003	○
	ACTDIP004	●
	ACTDIK008	○
	ACTDIP009	○
	ACTDIP010	●
	ACTDIP011	●
	ACTDIP012	●
	ACTDIP013	●
	ACTDIP017	●
	ACTDIP018	●
	ACTDIP019	●
	ACTDIP020	●
	ACTDIP021	○
	ACTDIP022	○
	ACTDIP027	●
	ACTDIP028	●
	ACTDIP029	●
	ACTDIP030	●

	ACTDIP031	<input type="radio"/>
	ACTDIP038	<input type="radio"/>
	ACTDIP039	<input type="radio"/>
	ACTDIP040	<input type="radio"/>
	ACTDIP042	<input type="radio"/>
Design & Technologies	ACTDEP005	<input type="radio"/>
	ACTDEP006	<input checked="" type="radio"/>
	ACTDEP008	<input type="radio"/>
	ACTDEP009	<input checked="" type="radio"/>
	ACTDEP015	<input type="radio"/>
	ACTDEP017	<input type="radio"/>
	ACTDEP018	<input checked="" type="radio"/>
	ACTDEP025	<input type="radio"/>
	ACTDEP026	<input type="radio"/>