

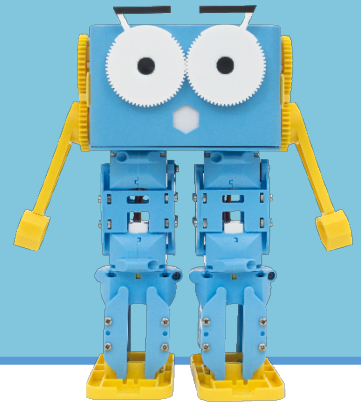
Lesson 1.13 – Teaching Marty How to Walk

Education Level: Second Level (Age 7-11)

Lesson Duration: 45 minutes

Prerequisite Knowledge: Lessons 1.1-1.12

Device Compatibility: Laptop, PC or Tablet



Lesson Overview

So far, students have been using the built-in command for getting Marty to take steps forwards and backwards. However, in this lesson we will create a program to get Marty taking steps without using the built-in command. Students will focus on breaking down how they walk in terms of balancing and where they shift their weight with each step.

Learning Objectives

- Breakdown and describe how we walk in terms of balancing and individual movements
- Program Marty to walk without using the built-in command
- Make use of functions, parameters, loops and if statements appropriately to create a walking program for Marty

Key Vocabulary

- Walking
- Joints
- Movement
- Balancing
- Weight
- Command
- Function

Resources & Equipment

- Marty the Robot
- Marty Workbook (Lesson 4)
- Laptops/Computers/Tablets
- Access to the Scratch editor

Additional Reading

- Educator's Guide
- Introduction to Programming with Marty using Scratch

Learning Plan & Activities

1. Start off by getting students to think about how they walk by breaking down the action into small steps and thinking about balance and whether they shift their body weight to a specific side
2. Can students program their friends to take one step forwards using only the instructions given in their workbooks/slides?
3. Moving into small groups with a device and a Marty, get students to try and translate the instructions they just gave each other to get Marty to take one step forwards with his right leg
4. Then again with his left leg
5. Ask students whether they can use something to stop repeating the blocks for each left and right step
 - a. Functions! Students should create two functions: one for stepping with the right leg and one for stepping with the left leg
6. Then create a loop that calls both of these, but do students notice how many steps Marty takes per loop or iteration?
 - a. He currently should be taking two steps each time we go around the loop but what if we only want to take 1 step? We need to keep track of the last foot we stepped with

- b. Using a variable, each time we call *step left* or *step right*, update the variable to hold either “left” or “right” and check to decide which function should be called
- c. Place this code in a function called *step*
- d. Then whenever we call *step* in our main sequence of blocks, we will be able to get just one step!

Additional Challenges

- See if students can create something similar to get Marty to kick either his left or right leg or any other actions that they can break down
- Get students to think up some yoga poses and see if firstly they think Marty will be able to do them and then recreate them in Scratch to get students to continue thinking about weights and balancing

Curriculum Benchmarks

Curriculum for Excellence – Technologies Benchmark Guide

● = Fully Addresses Benchmark ○ = Partially Addresses Benchmark

Curriculum Organiser	Benchmark Covered	Lesson 1.13
Digital Literacy	TCH 0-01a	●
Craft, Design, Engineering and Graphics	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 1.13
Computing	1-a	●
	1-b	●
	1-c	●
	1-e	●

	2-a	●
	2-b	●
	2-c	●
	3-a	●
	3-b	●
	3-d	○
	4-a	○
	4-b	○
Design & Technology	1.1-a	●
	1.3-b	●
	2.3-b	●
	3.1-b	○
	3.3-c	○

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

● = Fully Addresses Benchmark ○ = Partially Addresses Benchmark

Curriculum Organiser	Benchmark Covered	Lesson 1.13
Digital Technologies	ACTDIK001	●
	ACTDIK002	●
	ACTDIP003	○
	ACTDIP004	●
	ACTDIK008	●
	ACTDIP009	●
	ACTDIP010	●
	ACTDIP011	●
	ACTDIP012	●
	ACTDIP013	●
	ACTDIP017	●
	ACTDIP019	●
	ACTDIP020	●
	ACTDIP021	○
	ACTDIP027	●
	ACTDIP028	○
	ACTDIP029	●
	ACTDIP031	○
	ACTDIP038	○
	ACTDIP039	○
ACTDIP040	○	
ACTDIP041	○	
Design & Technologies	ACTDEK001	○
	ACTDEK002	○
	ACTDEK004	○
	ACTDEP005	○
	ACTDEP006	●

	ACTDEP009	<input checked="" type="radio"/>
	ACTDEK010	<input type="radio"/>
	ACTDEK013	<input type="radio"/>
	ACTDEP015	<input type="radio"/>
	ACTDEP016	<input type="radio"/>
	ACTDEP018	<input checked="" type="radio"/>