LEGO® Education WeDo
COMPUTING SCHEME OF WORK

LEGOeducation.co.uk

LEGO education

Introduction

The LEGO® Education team is pleased to present this computing scheme of work for use in primary education at key stage 1 (years 1-2) and key stage 2 (years 3-6). These materials will help you deliver an exciting computing project based around relevant technology, which will enable students to learn fundamental computing and programming concepts using a real-world context.

BACKGROUND TO THIS SCHEME OF WORK

The UK Government has announced that a new computing curriculum for key stages 1-4 is to be launched in September 2014.

This curriculum has a much stronger emphasis on programming and coding than ever before, especially at primary level. With this in mind, LEGO Education has produced this scheme of work for key stages 1 and 2 using LEGO® Education WeDo to help students tackle this abstract subject. The scheme consists of 12 lessons, which, depending on how they are organised, could equate to around 24 hours of classroom-based activity.

TARGET GROUP

These activities will help you inspire students to think about how important computer programming is in our everyday lives. Students will gain experience in basic programming, through a mixture of direct teaching, experimentation and exploration.

The material is aimed at key stages 1 and 2 and can be adapted for any primary school year groups. Whilst this material has primarily been written to address the computing curriculum, there are ample cross-curricular opportunities to impact other subject areas, most notably science, mathematics and design technology.

The lesson plans in this scheme of work will greatly reduce planning time, and the experience gained will help you produce further plans as your experience grows.

ORGANISATION OF THE SESSIONS

The lessons in this scheme of work are organised in line with LEGO Education's 4 Cs approach. In every lesson, students will have opportunities to Connect to the task and the bricks through a series of engaging activities. Every lesson requires students to Construct a variety of models, including their own designs. There are plenty of opportunities for the students to Contemplate the process through group discussion and specific questioning, details of which are included in the lesson plans. The students will be expected to Continue their learning in linked curriculum areas, through extension tasks and open-ended challenges.

Each lesson is usually split into four main activities, although where activities require additional time the lesson is split into two or three main parts. Activities include the use of LEGO® Education WeDo Software and hardware, as well as traditional teaching resources and ideas that teachers will be familiar with.

Guidance and solutions for activities are contained within the lesson plans. The lessons should include building time, when required.

Each activity should include a discussion phase with, and between, the students about the structure/design of the program and the terminology being used.

Introduction

USE OF THE INTERNET, INCLUDING VIDEO HOSTING WEBSITES

In this scheme of work, we have avoided providing direct links to search engines or video hosting sites, as they can become obsolete over time.

To overcome these restrictions, we have provided a number of keywords that you can enter into your preferred search engine to find appropriate content.

FEATURED MODELS

This scheme of work features six of the original twelve LEGO® Education WeDo models. The Roaring Lion and Hungry Alligator models (referred to as a Crocodile in the lesson plans) are used in the year 1 and year 2 lessons respectively. The Goal Kicker and Goal Keeper models are used in the lower key stage 2 lessons, while the Dancing Birds and Drumming Monkey models are used in the upper key stage 2 lessons. The rationale behind using these six models is that the lessons should provide the simplest builds and the best opportunities for designing and building simple programs.

MODELS NOT INCLUDED

Though six of the original LEGO® Education WeDo models have not been included in this scheme of work, you may consider allowing students to explore and test these models and programs, as they become more proficient. Refer to the LEGO® Education WeDo activities and Teacher's Guide for further details.

IDEAS FOR DOCUMENTING COURSEWORK AND CREATING A PORTFOLIO

Throughout this scheme of work, it is recommended to have the children document their research and findings. Recorded documentation may include:

- Assessment Resource Sheets
- Photos of student builds
- Photos of pairs/groups working together
- Videos of the builds and programs
- Drawings/diagrams of the builds
- Screenshots of the programs
- Handwritten or typed diary entries
- Handwritten or typed predictions and findings
- Written or drawn instructions
- Annotated resource sheets
- Handwritten or typed callouts
- Videos or audio recordings of pair/group/class discussions

Introduction

HOW TO USE DOCUMENTATION FOR CROSS-CURRICULAR PURPOSES

Although this scheme of work is designed for the new computing curriculum, it also documents a student's understanding across a range of curriculum areas. The activities carried out by the students require verbal and listening skills, problem solving, teamwork, and knowledge and understanding of mathematics, science, design technology, ICT, art and literacy.

Certain lessons included in this scheme of work could readily be used in a science investigation or to supplement design technology or ICT lessons.

RATIONALE BEHIND VOCABULARY USE

Each lesson has been designed to help you and the class use increasingly accurate vocabulary. While more experienced teachers may be tempted to adopt their own vocabulary, we recommend using this scheme of work to guide the level of discussion. As an example, early lessons refer to LEGO builds as models, although later on we introduce the term, hardware. This allows the class to become familiar with technical computing concepts before they understand and apply the correct terminology at a later point. Some will already be familiar with the technical vocabulary, and all children must be encouraged to use the correct terms as quickly and as frequently as possible.

Key vocabulary is included in a glossary at the end of this scheme of work, along with child-friendly definitions that you can share with the class. These words are underlined the first time they appear in each lesson.

RATIONALE BEHIND LESSON TIMINGS (SHORTER TO LONGER)

As with the vocabulary, each lesson is designed to help you and the class use increasingly technical approaches to computing and programming. Therefore, earlier lessons are split into four shorter activities to ensure structure and engagement. Later lessons are often split into longer activities, which allows for greater independence and open-ended tasks for older students. These longer activities can still be broken down into shorter sections using mini-plenaries. A series of questions for you to ask during mini-plenaries and ongoing assessments has been provided in the form of callouts located to the right of each lesson plan.

USING BUILDING INSTRUCTIONS

Most children are familiar with LEGO® building instructions. As they are easy to follow, it is recommended that you use the LEGO® Education WeDo instructions each time a build is necessary in a lesson. However, there are alternatives to using these every time. For example, you or a pair of students could pre-build a model needed for a lesson, which the class could then copy to create their own builds. This allows for a different set of skills during the building process. Some children may even be able to build a model from memory.

Whichever way you decide to run each activity, it is generally advised that you are familiar with the model to be built in each lesson and that you have pre-built it before the lesson starts. This model could also be used during the lesson as an example, especially for younger students.

SOLUTION EXAMPLES

Whenever a solution is given, either for a model or program, it is advisable to point out to the students that it is one solution. In most cases, more than one solution will be possible and undoubtedly the students will find their own that will work. Examples should be used to guide students to their own solution. Children will enjoy the challenge of finding other solutions when one is provided for them.

CURRICULUM GRID

The following grid shows how each lesson corresponds to the new computing programmes of study for key stages 1 and 2.

Pupils should be taught to:		Lesson											
		1.1	1.2	1.3	2.1	2.2	2.3	3.1	3.2	3.3	4.1	4.2	4.3
Key Stage 1	Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions	◆	◆	◆	◆	◆	◆						
	Create and debug simple programs			◆	◆	◆	◆						
	Use logical reasoning to predict the behaviour of simple programs					◆	◆						
Key Stage 2	Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts							◆	◆	◆	◆	◆	◆
	Use sequence, selection, and repetition in programs; work with variables and various forms of input and output								◆	◆	◆	◆	◆
	Use logical reasoning to explain how simple algorithms work and to detect and correct errors in algorithms and programs									◆	◆	◆	◆
	Select, use and combine a variety of software on a range of digital devices to design and create a range of programs												◆
Both	Use technology safely and respectfully	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆

MODEL GRID

The following grid shows the LEGO® Education WeDo models used in each lesson.

Model	Lesson											
	1.1	1.2	1.3	2.1	2.2	2.3	3.1	3.2	3.3	4.1	4.2	4.3
The Roaring Lion		◆	◆									
The Hungry Alligator (Crocodile)				◆	◆	◆						
The Goal Kicker							◆	◆	◆			
The Goal Keeper								◆	◆			
The Dancing Birds										◆	◆	◆
The Drumming Monkey											◆	◆

AN INTRODUCTION TO PROGRAMMING USING LEGO® Education WeDo

LEGOeducation.co.uk

LEGO education

KEY STAGE 1 – YEAR 1 – USING THE ROARING LION TO HELP NARRATE AESOP'S FABLE, 'THE LION AND THE MOUSE'

Each of the three lessons is designed to be 2 hours long, broken down into 4 activities of roughly 25-30 minutes. There is a short plenary activity in each lesson of roughly 10 minutes duration.

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
1.1 – Understand what algorithms are Recognise ways to ensure safety when using technology	1. Explain what an algorithm is. Why do we put clothes on in a certain order? What happens if we do not use an order? How did you learn to get dressed? Why is it important to learn?	'Lavi The Lion Finds His Pride' by Linda Dickerson 'If I Were A Lion' by Sarah Weeks 'Library Lion' by Michelle Knudsen 'Lions' by Catherine Ipcizade 'Everything Big Cats' by National Geographic Kids	Toy with clothing
	2. Explain why algorithms are used (provide an example). How did your partner guide you round? What was hard about the task? Did you bump into anything? If so, why? Did they use of specific words/phrases make the activity easier?		Blindfolds It may be advisable to set out an obstacle course in a hall or open space – playground equipment is ideal
	3. Share Aesop's Fable 'The Lion and the Mouse' with the class. What did we learn about lions? What else do you know about lions? What is a lion like? What do they do?		Key search terms: Lion, mouse, Aesop, fable Resource sheet A
	4. Define the rules for working with the LEGO® Education WeDo sets and write them down on the whiteboard or make a display poster. How do we use the LEGO bricks safely and properly? What should we do/not do with them? Have the class build a lion model in pairs. How do we know this model is a lion? It is recommended that you show the LEGO® Education WeDo Software to the class in preparation for the next lesson. It will help that they are familiar with the look and feel of the software interface and program blocks.		Whiteboard/poster paper LEGO® Education WeDo sets

An Introduction to Programming using LEGO® Education WeDo™

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
1.2 – Understand how algorithms are implemented as programs on digital devices Understand that programs work by following precise instructions	**1. Show the instructions for building the LEGO® Education WeDo Roaring Lion** How are the instructions written? Is it easy to follow without words? Why do they not use words? **Help the class build the Roaring Lion model using the building instructions.**		LEGO® Education WeDo Roaring Lion building instructions LEGO® Education WeDo sets Resource sheet B for introduction
	2. Explain what a program is. Show the LEGO® Education WeDo algorithm on the whiteboard or on poster paper. What is an algorithm? What do you think the symbols in the program blocks show? Why do they not use words? What is the difference between an algorithm and a program?		Resource sheet C Whiteboard/poster paper
	3. Connect the lion model to the software and show the class what the algorithm does. How does the program make the lion move? Which block shows the motor? What does the motor do? How does the motor make the lion sit up? **Establish with the class that the software tells the hardware to move.** How are these instructions like the blindfold game?		LEGO® Education WeDo Roaring Lion (suggest pre-built by teacher) LEGO® Education WeDo Software
	4. Recap the blindfold game from previous lesson. Ensure the instructions given are clear and helpful. **Tell the class to draw an algorithm using blank program blocks to instruct the student to move across the classroom. The class could define what each block would look like before drawing.** What was step one in the instructions? What was step two...? How did we help them avoid obstacles? How clear did the instructions have to be? Why do they need to be precise instructions? What symbols will you use in your program blocks?		Blindfold Resource sheet D

An Introduction to Programming using LEGO® Education WeDo™

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
1.3 – Create a simple program	1. Revisit Aesop's Fable 'The Lion and the Mouse' while helping the class follow the instructions to build the Roaring Lion. What do we remember from the story? What does the lion do? How did the program tell the lion to move? Are there any other movements we could add to the program?		LEGO® Education WeDo sets LEGO® Education WeDo Roaring Lion building instructions Key search terms: Lion, mouse, Aesop, fable
	2. This lesson will be 45 minutes Explain that the class will follow the example on Resource sheet E to create a simple algorithm. Show them how to add program blocks as a whole class. Then allow them to explore the software.What do you think the symbols in the program blocks show? What do you think each block in the algorithm does? Why is it important they are in this order? What would happen if block x was taken out of the algorithm? What might happen if you added block y?		Laptops/computers LEGO® Education WeDo software Resource sheet E
	3. This lesson will be 45 minutes Explain that the algorithm they have created is only part of the program needed to move the lion. Help the class where necessary to highlight the blocks they need to use. In pairs, the class try to create a second algorithm, using the first as an example, to make the lion sit down again. Can you create a second algorithm to make the lion sit down? What blocks will you need? Can you explore the blocks to find out if they will make the lion sit down? How accurately does the model recreate the movements of the lion in the fable? How will we attach the hardware to the software? Retell the story while the class executes the program to make the lion move.		Laptops/computers LEGO® Education WeDo Software Resource sheet F

KEY STAGE 1 – YEAR 2 – USING THE HUNGRY ALLIGATOR TO HELP NARRATE THE STORY 'THE CROCODILE AND THE MONKEY'

Each of the three lessons is designed to be 2 hours long, broken down into 3 or 4 activities of roughly 25-45 minutes. There is a short plenary activity in each lesson of roughly 10 minutes duration.

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
2.1 – Create a simple program using precise and unambiguous instructions	1. Show the LEGO® Education WeDo Hungry Alligator building instructions. How are the building instructions written? How precise are they? Do they leave any room for getting it wrong? What do you think the crocodile will do?	The Enormous Crocodile' by Roald Dahl 'Lyle, Lyle, Crocodile' by Bernard Waber 'Crocodile Encounters!' by National Geographic Kids	LEGO® Education WeDo Hungry Alligator building instructions LEGO® Education WeDo Construction sets
	2. Share the story 'The Crocodile and the Monkey' with the class. What did you learn from the story? What do you know about crocodiles? How do they move? How do they catch their prey?		Key search terms: Crocodile, monkey, India, jamun Resource sheet G
	3. This lesson will be 1 hour long. Review basic programming and LEGO® Education WeDo Software. Explain that the class will follow the example to create a simple algorithm. Then allow them to explore other program blocks in the software. Point out the Sensor block. What might the sensor do? How will the program make the model move? What predictions can you make?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers Resource sheet H

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
2.2 – Understand how to debug a simple program	1. Review that instructions need to be precise and unambiguous. Help the class so that they can follow the instructions to build the Hungry Alligator model in pairs. Review with them what they remember about motors and gears. How do the gears and motor make the crocodile's mouth open? What does the sensor do? Where else in real life would you find sensors?		LEGO® Education WeDo Construction sets
	2. Discuss with the class the similarities and differences between a set of algorithms. Which blocks do the algorithms have in common? Which ones are different? What do you think the different blocks do? Can you predict how they will affect the model when you execute the program? Question the class to find out if they can identify which algorithm opens and closes the crocodile's jaws.		Resource sheet I and J
	3. Review with the class how to connect hardware to the software. Explain that they must build each of the algorithms in the software and execute the program to test it. Which of the algorithms works correctly? Why do the others not work? Can you identify which program block is stopping the algorithm from working properly?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers
	4. Ask the class to pick one of the algorithms and attempt to add or replace program blocks to make it work properly. Tell them they are attempting to recreate the correct algorithm on Resource sheet I as closely as possible. Which program block needs to be replaced in the algorithm? Is there a block missing? How do you check if you have debugged the program properly? Does the program execute properly now?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers Resource sheet I

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
2.3 – Predict the behaviour of a simple program through logical reasoning Solve a problem through predicting and debugging a simple program	1. Explain that the class will build the crocodile model from previous lessons as independently as possible. Try to keep a hands-off approach to this activity. While they build, facilitate a discussion about debugging a simple program. What does debugging mean? How do you debug a simple program? Who has an example of what they changed or added to their algorithm in the previous lesson?		LEGO® Education WeDo Construction sets
	2. This activity will be 45 minutes long. Explain that the class will look at Resource sheet K and discuss similarities and differences between the algorithms. What is similar about each algorithm? Which blocks do they have in common? Which ones are different? How do you think the different blocks will change the program? What do you think each program will do? Can you predict how it will affect the movement of the hardware? Review any unfamiliar blocks with the class, using Resource sheet L. Ask the class to identify which algorithms they think will do specific jobs.		Resource sheet K and L
	3. Introduce the new learning objective to distinguish this activity from the last one. Explain that the class will need to select the algorithm they think will open and close the crocodile's jaws with a 'crunch' sound. Once selected, help them to debug the program so that it plays the sound successfully. Which algorithm do you think is most likely to open the crocodile's jaws? Can you spot if there are any particular programs with that algorithm? What if it does not execute exactly as you want it to? Which blocks might be causing the problem? What would you change or add to the algorithm to make the program execute properly?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers Resource sheet K and L

KEY STAGE 2 – YEAR 3 AND 4 – USING THE GOAL KICKER AND GOAL KEEPER MODELS AND SPORT IMAGES

Each of the three lessons is designed to be 2 hours long, broken down into 3 activities varying between 25 minutes and 1 hour in length. There is a short plenary activity in each lesson of roughly 10 minutes duration.

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
3.1 – Review how to create and execute simple programs Design a program with a specific goal	1. Review basic programming and LEGO® Education WeDo Software with the class. Explain that the class will build the LEGO® Education WeDo Goal Kicker model from the building instructions. What do you remember about programming? Can you give a definition for any of the following keywords: algorithm, program, hardware, software, execute, debug, motor, sensor? What does the Goal Kicker model do? How does it use the sensor?	'The David Beckham Academy Football Stories' 'Frankie's Magic Football' by Frank Lampard 'Hope Solo: My Story (Young Readers' Edition)'	LEGO® Education WeDo Construction sets LEGO® Education WeDo Goal Kicker building instructions
	2. Explain how the class connect the hardware to the software. Have the class create the algorithm on Resource sheet N. Discuss what the algorithm does. Can they identify which program blocks make the program execute in this way? Which of the program blocks are familiar? What do you think it will make the model do? Which blocks make the program execute in this way? How is the sensor involved? What does the motor do?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Resource sheet N
	3. This activity will be 1 hour long. Introduce the new learning outcome: designing a program with a specific goal. Explain to the class that they will need to design a new program to achieve a specific goal, possibly by debugging the previous algorithm. The class, or smaller groups, could define that goal. Examples may include having the kicker score in a specific goal or changing the height/angle of the kick. How can you define a specific goal for the hardware to achieve? How will you design the program? Could you use the suggested algorithm and modify it?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
3.2 – Explore how to use sequence, selection and repetition in a program	1. This activity will be 45 minutes long. Explain that the class will build the Goal Kicker model featured in previous lessons as independently as possible. Try to keep a hands-off approach to this activity. Explain that they will need to build the algorithm from Resource sheet N used in the previous lesson to program the Goal Kicker. What do we mean by independent? How can you work together to solve your own issues? Who else could you ask before me? How have you identified and solved the issues?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Goal Kicker building instructions LEGO® Education WeDo Software Laptops/computers Resource sheet N
	2. This activity will be 45 minutes long. Help the class to build the Goal Keeper model from the building instructions. Then use the second algorithm on Resource sheet N to program the Goal Keeper. How is the Goal Keeper program different to the Goal Kicker? Which program blocks are different? Why do the two algorithms need to be different? How might you be able to combine the two models and algorithms?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Goal Keeper building instructions LEGO® Education WeDo Software Laptops/computers Resource sheet N
	3. Partner up the pairs of students into groups of four. Explain how sequence, selection and repetition are used in programming to create a loop for hardware to continue to interact without stopping or resetting. Use Resource sheet O to highlight each element and question the class. Which program blocks show that the algorithm will be repeated? Which program blocks have been selected to do a particular job? Are you familiar with any of these selected blocks? Have the groups of four discuss what algorithm they would need to define in order to make the Goal Kicker and Goal Keeper interact together repeatedly. Will they be able to create one program that controls both models? How might you get the two models to interact? What would the Kicker do? What would the Keeper do? How would the algorithms need to be designed to allow the two models to interact?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers Resource sheet O

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
3.3 – Design and debug a program with a specific goal Solve a problem by breaking it down into smaller parts	1. Explain to the class that they will build the two models (one per pair) in the groups they were in for the previous lesson. Show them the algorithms on Resource sheet N for the two models and have them enter, connect and execute the program. What do you want the two models to do? How should they interact? What might the implications be for the program?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Goal Kicker and Goal Keeper building instructions LEGO® Education WeDo Software Laptops/computers Resource sheet N
	2. This activity will be 1 hour long. Explain to the class that they will need to solve a problem: to design and debug a program with a specific goal. Help the class to make their two models interact. Allow the class to explore the different program blocks. Guide the class towards the example solution programs on Resource sheet Q. This can be shared with the class at any point during the lesson. What have you done so far? What program blocks have you explored? How are the two models interacting? How is everyone in your group involved in the designing and debugging process? What is your specific role? Given more time, what else would you like the program to do? What are the most difficult parts of solving this problem?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers Resource sheet Q
	3. Explain that each group will present their models and programs to the rest of the class. These presentations will be short explanations of what they wanted the models to do, which blocks they selected and how they attempted to create a sequenced, repeating program. Resource sheet R can be introduced as a way for the class to assess their own work during this lesson. Did you solve the problem? If so, how? What were the difficult parts of solving the problem? How did you apply what you had learnt about programming? What were you most successful at as a group? What were you most successful at as an individual?		Resource sheet R

KEY STAGE 2 – YEAR 5 AND 6 – USING AMAZING MECHANISMS (LINK TO CAMS/DT)

Each of the three lessons is designed to be 2 hours long, broken down into 2 or 3 activities of between 25 minutes to 1 hour in length. There is a short plenary activity in each lesson of roughly 10 minutes duration.

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
4.1 – Review how to design and debug simple programs Explore working with variables	1. 1. This activity will be 1 hour long. Review basic programming and the LEGO® Education WeDo Software with the class. How do the mechanical toys work? What do gears, pulleys, cams and levers do? What do you remember about programming? Can you recall the definition of any of the following key words: algorithm, program, hardware, software, execute, debug, motor, sensor? Explain that the class will follow the instruction to build the LEGO® Education WeDo Dancing Birds model and connect it to the software. Have the class create the algorithm on Resource sheet T, then test it and debug the program to make it work properly. What did you notice was wrong about the algorithm? What have you done so far? What program blocks have you explored? Have you been able to use sound?	'How to Design and Make Simple Automata' by Robert Addams 'Making Moving Toys' by Pippa and Ian Howes	LEGO® Education WeDo Construction sets LEGO® Education WeDo Dancing Birds building instructions LEGO® Education WeDo Software Laptops/computers Resource sheet S and T
	2. This activity will be 1 hour long. Define this period as a task for which the class will determine the outcome, and define the boundaries of the outcome using Resource sheet T. Explain what variables are and question the class throughout the activity to assess their understanding. Which variables have you worked with? What have you observed has happened to the birds as you apply the changes? Does the program work better with the changes? If so, why? What else could you change, if you had more time?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Software Laptops/computers Resource sheet T

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
4.2 – Design and debug a program Explore how to combine programs, using variables, with a specific goal	1. This activity will be 1 hour long. Discuss how the class think the LEGO® Education WeDo Drumming Monkey model will work, based on the instructions. What do you think the monkey will do? What program blocks will be included in the algorithm? Can you justify your predictions for which program blocks will be included in the algorithm? Explain that the class will follow the instructions to build the LEGO® Education WeDo Drumming Monkey and connect it to the software. Using Resource sheet U, explain that the class will explore the program blocks to design a new algorithm that makes the monkey plays different beats. Discuss what their new algorithm does by sharing experiences across the class. What have you changed in the algorithm? How has it affected the monkey? What needs to change to make the new algorithm on Resource sheet U?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Drumming Monkey building instructions LLEGO® Education WeDo Software Laptops/computers Resource sheet U
	2. This activity will be 1 hour long. After forming the pairs into groups of four, review how sequence, selection, repetition and variables are used in programming. Show the examples on Resource sheet V. How can you apply your knowledge of sequence, selection, repetition and variables? Which blocks show that the algorithm will be repeated? Which blocks show variables that you can explore? Explain that one pair will build the Dancing Birds model, while the other pair will build the Drumming Monkey. As a group, they will design and debug a program that combines the algorithms for both models. Prompt them to share their experiences with the class by the end of the lesson. How might you get the two models to interact in new ways? How would the algorithms need to be designed to allow the two models to interact? Can the monkey create the beat for the birds to dance to?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Drumming Monkey and Dancing Birds building instructions LEGO® Education WeDo Software Laptops/computers Resource sheet V

Lesson and NC Requirement	Teacher input and questions	Suggested reading	Resources and links
4.3 – Design a program and detect and correct errors it Solve a problem, including documentation of a written program	1.This activity will be 1 ½ hours long. Review Resource sheets W, X and Y with the class. Tell them that they will guide the class through the documentation and self-assessment process. What do you want the hardware to do? How did you decide they should interact? What might the implications be for the program? Explain that in pairs, the class will be designing a program using either or both the Dancing Birds and Drumming Monkey. Tell the class that they will decide which models to make and explore the different program blocks for themselves. They will have free rein over what they create. What have you done so far? What program blocks have you explored? How is the hardware interacting? How did you assign roles within your pair or group? Given more time, what else would you like the program to do? What are the most difficult areas of solving this problem? Have you learnt anything new about programs and programming?		LEGO® Education WeDo Construction sets LEGO® Education WeDo Drumming Monkey and Dancing Birds building instructions LEGO® Education WeDo Software Laptops/computers Resource sheet W, X and Y
	2. Explain that each group will present their models and programs, whatever state they are in after the first activity. Reinforce that a 'finished' product is not essential. Tell them that they can use Resource sheets W, X and Y to help them make a short presentation to the class about what they did. What did you want the hardware to do? How did you plan for the two models to interact? What were the difficult parts of solving the problem? How did you apply what you had learnt about programming? What were you most successful at as a pair or group? What were you most successful at as an individual? How would you improve the process next time?		Resource sheet W, X and Y

SESSION 1.1

The Roaring Lion

LEGO education

Session 1.1
Lesson overview

This lesson is designed to introduce the class to the basic elements of computing and programming – algorithms. Through games and stories, the class will discover what an algorithm is and begin a journey towards creating a working LEGO® model of a lion.

During the lesson, you will read Aesop's Fable, 'The Lion and the Mouse' to the class.

LEARNING OUTCOMES

- Understand what algorithms are
- Recognise ways to ensure safety when using technology

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain what an algorithm is
- Describe ways in which the technology can be used safely
- Describe their LEGO lion model

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a lion using LEGO bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is important to point out the expectations of using the set with the class, for example, making sure that sets are put away just as they receive them.

ABOUT THE SOFTWARE

The LEGO® Education WeDo Software is not used during this introductory lesson, instead it focuses on engaging the class and familiarising them with the basics of computing and programming language.

However, it is recommended that you are familiar with the software, prior to using it during the next lesson.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are four main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 1.1
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: a toy with clothing, blindfolds, an interactive whiteboard or poster paper, a version of Aesop's Fable (Resource sheet A), and LEGO® Education WeDo Construction Sets.

Session 1.1
Lesson Plan

INTRODUCTION

Pose the question, What are you wearing?, followed by, How did you get dressed?

For example: Firstly, I had to take my underwear from the drawer and put it on, including both my socks. Then I took my shirt off the hanger and put it on. I did the buttons up from the bottom.

Understand what algorithms are.

MAIN ACTIVITY 1

- Take some feedback from the class about what they are wearing and how they got dressed. This questioning is setting up the idea of a set of items being put on in a particular order to create the right outcome.
- Explain that computers work in the same way. They need a set of instructions that show them what to do in the right order. Remind them that computers, unlike humans, do not have the ability to think for themselves. They only do what a person has told them to do.
- Explain what an algorithm is. Ensure the class understand that a computer needs to be given an algorithm by a person, otherwise it will not know what the person wants it to do.
- Dress a toy or teddy bear by asking the class what clothing should go on and in what order. Discuss what the 'correct' order might be to create the right outcome. You may wish to intentionally dress the teddy in the incorrect way to highlight how a mistake in the instructions can lead to an undesired outcome.

Why do we put clothes on in a certain order?

What happens if we do not use an order?

How did you learn to get dressed? Why is it important to learn?

MAIN ACTIVITY 2

- Explain that algorithms are used so computers do not get confused and know exactly what you want them to do.
- In pairs, the class will guide each other around the classroom or another space. One is blindfolded or has their eyes closed, while the other gives them instructions. It may be advisable to set out an obstacle course in a hall or open space – safe playground equipment is ideal. Always remember that the safety of the student is paramount.
- Discuss how effective the instructions were that the class were giving to their partners. Have them offer each other advice to improve their instructions.
- Repeat the blindfold activity by changing roles. This time, tell the class they should only use specific words/phrases such as backwards, forwards, up, down, turn left, turn right, take three steps.
- Reflect with the class whether this made the activity easier.

How did your partner guide you round?

What was hard about the task?

Did you bump into anything? If so, why?

Did the use of specific words/phrases make the activity easier?

Session 1.1
Lesson Plan

MAIN ACTIVITY 3

- Use the key search terms: lion, mouse, Aesop, fable. Alternatively, use the version of the story found on Resource sheet A.
- Share Aesop's Fable, 'The Lion and the Mouse' with the class. Then divide the class into pairs and have them talk about what they learnt from the story.
- In pairs, have the class discuss what they already know about lions and share some key facts with the class. To prepare them for a future lesson, they will need to identify that lions lie down, sit up, are powerful and roar.
- Tell the class that they will be using this story in the last session and that they will be building using LEGO® bricks and designing an algorithm to make a lion model move.

What did we learn about lions?

What else do you know about lions?

What is a lion like? What do they do?

Recognise ways to ensure safety when using technology.

MAIN ACTIVITY 4

- With input from the class, define a class set of rules for working with the LEGO® Education WeDo Construction Sets. Write them down on the board or make a display poster.
- Distribute one LEGO® Education WeDo set to each pair.
- Tell the class to build their own lion using only the bricks in their own set. There must be no swapping or lending of bricks between pairs.
- Have the pairs share their models in small groups or in front of the class. Challenge the class to spot what makes each one a good lion.
- Explain to the class that they will be using a similar lion model to help them tell Aesop's Fable in a future session.
- It is recommended that you show the LEGO® Education WeDo Software to the class in preparation for the next lesson. It will help that they are familiar with the look and feel of the software interface and program blocks.

How do we use the LEGO bricks safely and properly?

What should we do / not do with them?

How do we know this model is a lion?

PLENARY

Have the class discuss what an algorithm is, revisiting the blindfold activity to refresh their memories if required.
Can they describe what it is and what it does? Are they using the definition you used earlier or can they put it into their own words? You are looking for all children to be able to do this confidently.

Before the next lesson, set them the challenge of explaining what an algorithm is to an adult.

An answer like, 'An algorithm is a set of instructions that makes a computer do something specific', demonstrates that they understand.

AESOP'S FABLE 'THE LION AND THE MOUSE'

Under a tall tree in the huge jungle, a small mouse quietly crept up to the sleeping lion. He was snoring loudly. The mouse felt safe while the lion slept. She studied the big cat's ears, his long whiskers and magnificent mane.

The little mouse felt very brave and pulled herself up onto the lion's tail. She decided the lion would make a fun slide and ran across his back, sliding down his front leg and off his large paw. This tickled the lion and he woke with a start, grabbing the mouse in his claws.

"Oh no!" exclaimed the mouse, "Please, Mr Lion, if you let me go I promise to come back and help you."

"Such a small, little mouse," the lion smiled, showing his sharp teeth, "how could you help me?"

The lion was roaring with laughter so hard that the mouse was able to escape from his claws and run out of the jungle.

The very next day the lion was returning home after a successful hunt. His tummy was full and he was sleepy. So sleepy that he didn't see the hunter's trap! He stepped into the rope and was immediately trapped. He growled fiercely, pulled and pulled, but could not free himself.

The mouse, hiding nearby, heard the roar and came to help, just as she promised. She noticed the rope that was holding the lion. She squeaked fiercely, nibbled and nibbled, and managed to free the lion!

The mighty lion looked down at the tiny mouse, "Oh, little mouse, how wrong I was to think you could not help me. I am sorry I laughed at you for being so small. You have saved my life and I shall protect you forever!"

SESSION 1.2

The Roaring Lion

LEGOeducation.co.uk

LEGO education

Session 1.2
Lesson overview

This lesson introduces students to model building using LEGO® instructions, and what algorithms look like in the LEGO® Education WeDo Software. The lesson will then progress to observing what a program does when connected to the model.

During the lesson, you should refer back to Aesop's Fable, 'The Lion and the Mouse'.

LEARNING OUTCOMES

- Understand how algorithms are implemented as programs on digital devices
- Understand that programs work by following precise instructions

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain what a program is
- Give an example of when instructions are not clear and the consequence

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a lion using LEGO bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo Roaring Lion building instructions and have pre-built the lion model in advance of the lesson.

ABOUT THE SOFTWARE

The LEGO® Education WeDo Software is introduced in this lesson, although should only be used by you as a whole-class activity.

It is recommended that you are familiar with the LEGO® Education WeDo Software and have pre-built the algorithm on Resource sheet C in the software in advance of the lesson.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are four main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 1.2
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: Resource sheet B, C and D, an interactive whiteboard or poster paper, LEGO® Education WeDo Construction Sets, LEGO® Education WeDo Software, and blindfolds.

Session 1.2
Lesson Plan

INTRODUCTION

Look at the pictures on Resource sheet B and discuss what they tell us about lions. This is a good activity to promote deducing information from a picture and the use of powerful adjectives

Understand how algorithms are implemented as programs on digital devices.

MAIN ACTIVITY 1

- Using the software, show the building instructions for building the LEGO® Education WeDo Roaring Lion. Discuss the instructions with the class and ask them if they find them easy to follow. Identify if there are students in the class who are familiar with LEGO building instructions.
- In pairs, help the students build the lion model by following the building instructions.
- Refer the class back to the models they made in the previous lesson. Are they similar to the one they are building today?

How are the instructions written?

Is it easy to follow without words?

Why do they not use words?

MAIN ACTIVITY 2

- Explain what a program is and how it relates to the concept of an algorithm. Show the LEGO® Education WeDo algorithm on the whiteboard or on poster paper.
- Ask the class to cut out the large program blocks from the bottom of Resource sheet C in pairs, and arrange them so they match the completed version on the board (also shown at the top of Resource sheet C).
- It is important in this activity that the class can distinguish the difference between an algorithm and a program. The glossary at the end of this book gives examples of how this can be explained.
- Students should glue down the program blocks in the correct order onto a new sheet of paper. Those that finish quickly can help the others ensure their algorithm matches the example.
- Tell the class that this is the algorithm they will using in this lesson to make the lion move.

What is an algorithm?

What do you think the symbols in the program blocks show?

Why do they not use words?

What is the difference between an algorithm and a program?

Session 1.2
Lesson Plan

MAIN ACTIVITY 3

- Connect the lion model to the computer as shown in the LEGO® Education WeDo manual.
- Ask the class to follow what the lion does as you execute the program containing the algorithm. Observe and discuss what is happening to the lion.
- Ensure they understand that the model moves because the program is making a motor work within the model. Ask the class to point to the block in their algorithm that shows the motor.
- Establish with the class that the software tells the hardware to move. Discuss that the software tells the hardware to move, by issuing it a set of instructions, exactly like the class did with each other in the blindfold game in the previous lesson.

Understand that programs work by following precise instructions.

How does the program make the lion move?

Which block shows the motor?

What does a motor do?

How does the motor make the lion sit up?

How are these instructions like the blindfold game?

MAIN ACTIVITY 4

- Recap the blindfold game from the previous lesson. Remind the class how clear and helpful the instructions given had to be (precise is a better term).
- One suggested activity is to blindfolded one student and have another student guide them across the classroom, while the rest of the class watches, listens and gives input if needed.
- Show the class the blank program blocks on Resource sheet D. Divide the class into pairs and ask them to draw an algorithm using the blank blocks that instructs a student to move across the classroom. Consider asking the students to define what each block/movement would look like as a symbol, before they begin drawing.

What was step one in the instructions?

What was step two...?

How did we help them avoid obstacles?

What symbols will you use in your program blocks?

Why do they need to be precise instructions?

PLENARY

Have the class discuss what a program is. Ask them to explain how a program uses an algorithm to make hardware move.

Set them the challenge of thinking about all the sets of instructions they come across in their daily lives. They can feed back with examples of these before or at the start of the next lesson.

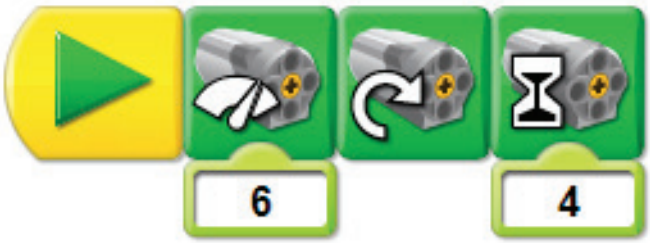
Examples might include: making a cup of tea, following cooking recipes, learning to count, brushing their teeth, playing a board game or following a knitting pattern.An answer like, 'An algorithm is a set of instructions that makes a computer do something specific', demonstrates that they understand.

Session 1.2
Resource sheet B

LIONS

Session 1.2
Resource sheet C

POSSIBLE ALGORITHM



PROGRAM BLOCKS TO CUT OUT


Motor On For Block

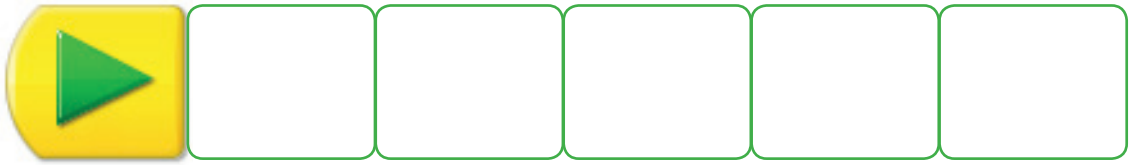

Motor Power Block


Motor This Way Block


Start Block

Session 1.2
Resource sheet D

BLINDFOLD GAME ALGORITHM



SESSION 1.3

The Roaring Lion

LEGOeducation.co.uk

LEGO education

Session 1.3
Lesson overview

This lesson is designed to get the class creating their own simple programs. This will include exploring various program blocks within the software, as well as creating a predetermined algorithm. This lesson has only three activities, with main activities 2 and 3 each requiring 45 minutes to complete. At the beginning of the lesson, you will refer back to Aesop's Fable, 'The Lion and the Mouse'.

LEARNING OUTCOMES

- Understand and apply knowledge of how to create a simple program

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain how a simple program works
- Create a simple program

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a lion using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo Roaring Lion building instructions and have pre-built the lion model in advance of the lesson.

ABOUT THE SOFTWARE

Following on from the previous lesson, the class will be using the LEGO® Education WeDo Software in this lesson to begin creating a simple program.

It is recommended that you are familiar with adding program blocks in the LEGO® Education WeDo Software.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts. These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 1.3
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: Resource sheet E and F, Aesop's Fable, LEGO® Education WeDo Construction Sets, LEGO® Education WeDo Software, and laptops/computers.

Session 1.3
Lesson Plan

INTRODUCTION

Have the class give directions from their class to another room in the school (hall, office, library). Discuss how specific they need to be with their instructions and why this is.

Create a simple program.

MAIN ACTIVITY 1

- Revisit Aesop's Fable, 'The Lion and the Mouse'. This can be done as a class activity, with the class contributing what they remember of the story while completing the following activity.
- In pairs, help the class follow the LEGO® Education WeDo building instructions to build the Roaring Lion.
- Discuss how the program made the lion sit up and what else the lion does in the fable.
- The class may observe that the lion will need to sit down again.

What do we remember from the story?

What does the lion do?

How did the program tell the lion to move?

Are there any other movements we could add to the program?

MAIN ACTIVITY 2 (45 MINUTES)

- Explain that the class will follow the example on Resource sheet E to create a simple algorithm.
- Show them how to add program blocks as a whole class to create the example on Resource sheet E.
- In pairs, have the class follow the example to add program blocks to recreate the algorithm that makes the lion sit up.
- Then allow them to explore the software further, perhaps adding other blocks. However, at the end of the lesson, each pair should have the example algorithm ready for when they attach the hardware.
- It is important that the class makes predictions about how the program will affect the hardware.
- Students who finish quickly could help the others complete the activity.

What do you think each block in the algorithm does?

Why is it important they are in this order?

What would happen if block x was taken out of the algorithm?

What might happen if you added block y?

Session 1.3
Lesson Plan

MAIN ACTIVITY 3 (45 MINUTES)

- Explain to the class that the algorithm they have created is only part of the program needed to move the lion.
- In pairs, ask the class to try and create a second algorithm, using the first as an example, to make the lion sit down again.
- Where necessary, help the class to identify the blocks they need to use.
- If they haven't already done so, allow the class to explore the variety of blocks and, if needed, guide them to the most useful blocks for this activity.
- Use the example on Resource sheet F as one solution. This can be shared with the class or in pairs.
- Students who complete this can be challenged to include sound in the algorithm to make the lion roar, or become advisors to other groups. The roar is Sound Block 14, as shown in the second algorithm on Resource sheet E.
- Towards the end of this lesson, attach the models to the computers (hardware to software) and execute the two algorithms to see if the models move as expected.

PLENARY

Retell the fable of 'The Lion and the Mouse'. Have the class execute their program to make the lionm ove at the right point in the story.

By making the lion move, discuss how the program enhanced the story.

Can you create a second algorithm to make the lion sit down?

What blocks will you need?

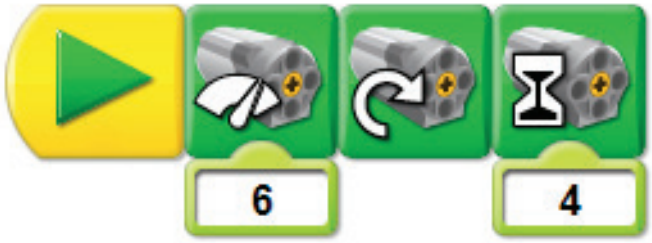
Can you explore the blocks to find out if they will make the lion sit down?

How accurately does the model recreate the movements of the lion in the fable?

How will we attach the hardware to the software?

Session 1.3
Resource sheet E

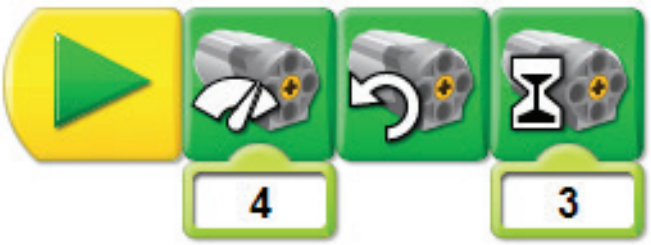
LION SITTING UP ALGORITHM



LION SITTING UP ALGORITHM WITH SOUND BLOCK

Session 1.3
Resource sheet F

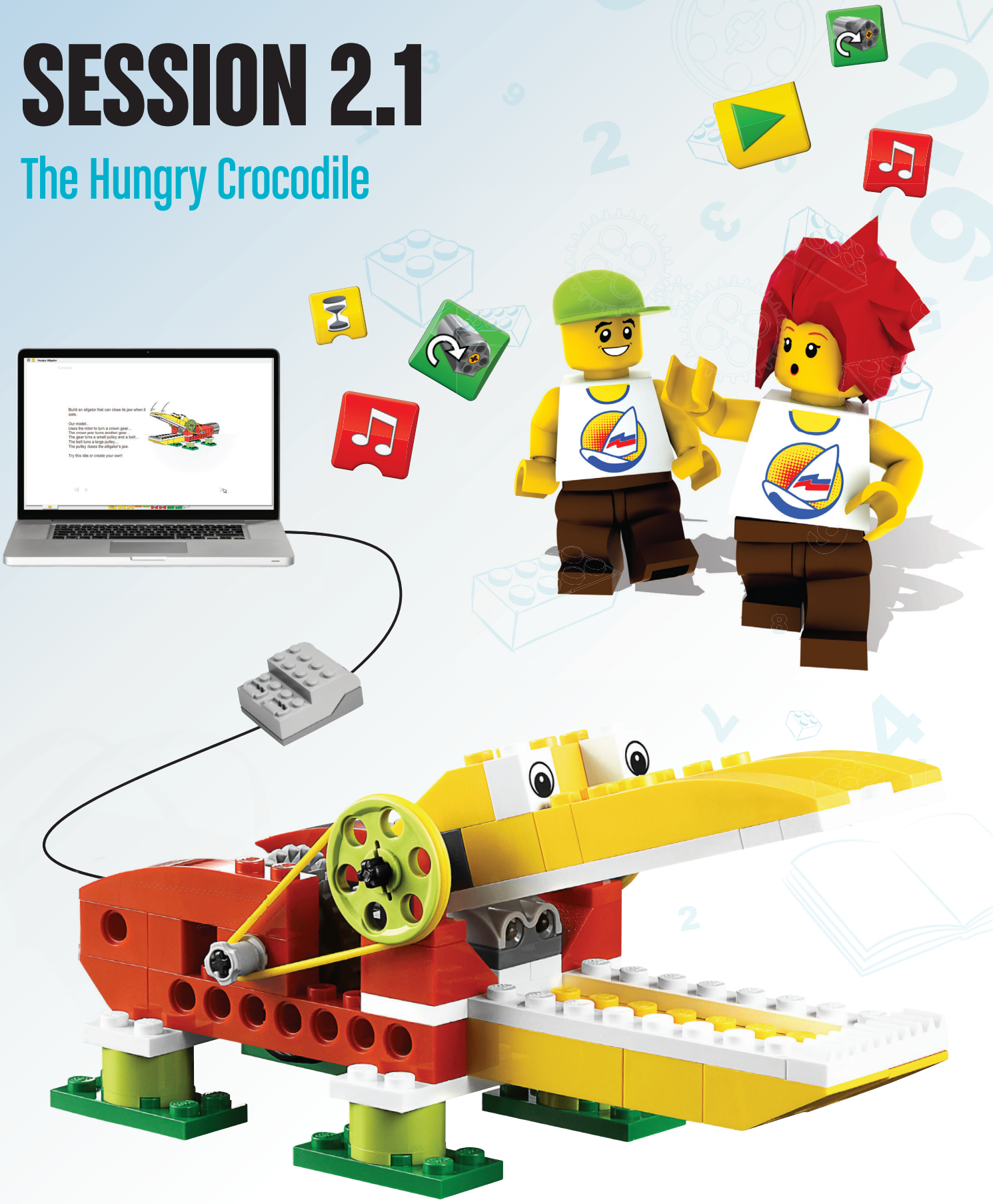
LION SITTING DOWN ALGORITHM SOLUTION



Sound Block 13 (Zzzz) could be added to this algorithm in the same way as the lion roar.

SESSION 2.1

The Hungry Crocodile



LEGO education

Session 2.1
Lesson overview

This lesson is designed to reintroduce the class to the basic elements of computing and programming – algorithms. The class will then be encouraged to create a simple program using precise and unambiguous instructions. Finally, the students will be given an hour to review what they have learnt about algorithms and experiment with the Sensor Block. During the lesson, you will read 'The Crocodile and the Monkey'.

LEARNING OUTCOMES

- Review what algorithms and programs are and how they are related to each other
- Create a simple program

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain what an algorithm is
- Explain what a program is
- Describe what their simple program is doing

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a crocodile using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is important to point out the expectations of using the set with the class, for example, making sure that sets are put away just as they receive them.

ABOUT THE SOFTWARE

The use of the LEGO® Education WeDo Software to create basic programs is introduced in this lesson.

It is recommended that you are familiar with the software, in particular the various program blocks, so that you can encourage the class to explore them when creating their own programs.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 2.1
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, building instructions and Software, Resource sheet G and H, and laptops/computers.

Session 2.1
Lesson Plan

INTRODUCTION

Review what an algorithm and a program are. Ask the class what the difference is between these two key words

Create a simple program using precise and unambiguous instructions.

MAIN ACTIVITY 1

- Show the class the building instructions for building the LEGO® Education WeDo Hungry Alligator.
- Discuss how precise the building instructions are and how they are designed to avoid mistakes when constructing a model.
- In pairs, have the class follow the building instructions to build the Hungry Alligator. Tell the class that while the LEGO® Education WeDo building instructions call it an 'alligator', you will be referring to it as a crocodile to fit the story you have chosen.
- Ask the class what they think the crocodile model will do while they are building it. What elements does the hardware use? Point out the motor, the gears and the pulley and belt.

How are the building instructions written?

How precise are they?

Do they leave any room for getting it wrong?

What do you think the crocodile will do?

MAIN ACTIVITY 2

- Use the key search terms: crocodile, monkey, India, jamun. Alternatively, use the version of the story found on Resource sheet G.
- Share the story, 'The Crocodile and the Monkey' with the class. Then divide the class into pairs and have them talk about what they learnt from the story.
- Still in pairs, have the class discuss what they already know about crocodiles and share some key facts with the class. Discuss what they know.
- Tell the class that they will design a program that will make the crocodile model move like it did in the story.

What did you learn from the story?

What do you know about crocodiles?

How do they move?

How do they catch their prey?

Session 2.1
Lesson Plan

MAIN ACTIVITY 3 (1 HOUR)

- In this activity, the class will be reviewing what they already know about basic programming using the LEGO® Education WeDo Software. Tell them that will be using a new element called a sensor.
- Explain that the class will follow the example on Resource sheet H to create a simple algorithm.
- Review with them how to add program blocks to create the example on Resource sheet H.
- In pairs, have the class follow the example to add program blocks to recreate the algorithm that makes the crocodile open and close its jaw.
- Then allow them to explore the software further, perhaps adding other blocks, such as Sound Block 17, (Crunch). However, at the end of the lesson, each pair should have the example algorithm ready for when they attach the hardware.
- It is important that the class makes predictions about how the program will affect the hardware.
- When students are ready, connect the crocodile model to the software and execute the program.
- Have the students observe and discuss what the crocodile does when the program runs.
- Students who finish quickly could help others complete the activity.

What might the sensor do?

How will the program make the model move?

What predictions can you make?

PLENARY

Have the class discuss what the algorithm did and how the program affected the hardware. Can they describe how the motor made the gears move the LEGO model? Can they describe how they think the sensor works?

Set them the challenge of explaining to an adult before the next lesson what their model did and how the motor and gears worked together.

What does the crocodile do?

What is the sensor's role in the program?

TRADITIONAL INDIAN TALE, 'THE CROCODILE AND THE MONKEY'

There once lived a very clever monkey. He lived in a tree that bore juicy, red apples. He was a very happy monkey. One fine day, a crocodile swam up and told the monkey that he was very hungry and asked nicely if he could have some food. The monkey kindly offered him three juicy apples. The crocodile enjoyed them and asked the monkey whether he could come again for some more fruit. The generous monkey happily agreed.

The crocodile returned the next day. And the next. And every day for the rest of the week. The two animals soon became good friends. The crocodile told the monkey about his wife and how they lived on the other side of the river. The kind monkey offered him some extra juicy apples to take home with him. The crocodile's wife loved the juicy apples and made her husband promise to get her some every day.

Meanwhile, the friendship between the monkey and the crocodile got stronger and stronger as they spent more time together. The crocodile's wife was very jealous. So, to end their friendship, she pretended that she could not believe that her husband could be friends with a monkey. Her husband tried to convince her that he and the monkey shared a true friendship. The crocodile's wife thought to herself that if the monkey lived on a diet of juicy apples, he would be very sweet to eat. So she asked the crocodile to invite the monkey to their house for dinner.

The crocodile was not happy about this. He tried to make the excuse that it would be difficult to get the monkey across the river. But his wife was so determined to eat the monkey that she thought of a cunning plan. One day, she pretended to be very ill and told the crocodile that the doctor said that she would only recover if she ate a monkey's heart! If her husband wanted to save her life, he must bring her his friend.

The crocodile was shocked. What could he do? On the one hand, he loved his friend. On the other, he could not possibly let his wife die.

So the crocodile went to the apple tree and invited the monkey to come home for dinner. He told the monkey that he could ride across the river on the crocodile's back. The monkey happily agreed. As they reached the middle of the river, the crocodile began to sink. The frightened monkey asked what was happening. The crocodile explained that he would have to kill the monkey to save his wife's life. The clever monkey told him that he would gladly give up his heart to save the life of the crocodile's wife, but he had left his heart behind in the apple tree. He asked the crocodile to be quick and turn back so that the monkey could get his heart from the apple tree.

The silly crocodile quickly swam back to the apple tree. The monkey scampered up the tree to safety. He told the crocodile to tell his wicked wife that she had married the biggest fool in the world.

Session 2.1
Resource sheet H

CROCODILE MOVEMENT ALGORITHM

SESSION 2.2

The Hungry Crocodile

LEGOeducation.co.uk

LEGO education

Session 2.2
Lesson overview

This lesson is designed to introduce the class to the process of debugging a simple program. They will have to choose and rationalise between a set of different algorithms, some of which are incorrect for the task they need to perform.

During the lesson, you can refer back to 'The Crocodile and the Monkey'.

LEARNING OUTCOMES

- Create and debug a simple program
- Review that programs work by following precise instructions

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain the rational behind debugging a simple program
- Give an example of how they debugged and improved an algorithm

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a crocodile using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo Hungry Alligator building instructions and have pre-built the model in advance of the lesson.

ABOUT THE SOFTWARE

The LEGO® Education WeDo Software is used in this lesson.

It is recommended that you are familiar with the LEGO® Education WeDo Software and have pre-built the algorithm on Resource sheet C in the software, in advance of the lesson.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 2.2
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, building instructions and Software, Resource sheet I and J, and laptops/computers.

Session 2.2
Lesson Plan

INTRODUCTION

Play a familiar game or sing a song with incorrect instructions.

For example, sing the song, Head, Shoulders, Knees and Toes, but mix up or leave out some of the body parts. Ask them what is wrong with this version and how confusing it is to follow.

Debug a simple program.

MAIN ACTIVITY 1

- Review how instructions need to be precise and unambiguous by showing the LEGO® Education WeDo Hungry Alligator building instructions.
- Divide the class into pairs and have them build the model using the building instructions.
- Review what they remember about motors and gears and how the algorithm from the previous lesson was designed to open and close the crocodile's mouth. Review also how the sensor reacts to something being put in front of it.

How do the gears and motor make the crocodile's mouth open?

What does the sensor do?

Where else in real life would you find sensors?

MAIN ACTIVITY 2

- Hand out Resource sheet J, which shows a series of similar algorithms. Discuss the similarities between them. Which blocks do the algorithms all have in common? Which ones are different?
- Discuss with the students what they think the symbol on each block shows and, therefore, what they think the block will do. Resource sheet I includes a short description of each block's function.
- One of the algorithms will make the crocodile open it's jaws and close them again when the sensor is triggered.
- Ask the class if they can identify which algorithm is the correct one. Resource sheet I has the correct algorithm.

The correct algorithm is also shown below.

Which blocks do the algorithms have in common?

Which ones are different?

What do you think the different blocks do?

Can you predict how they will affect the model when you execute the program?

Session 2.2
Lesson Plan

MAIN ACTIVITY 3

- Explain to the class that the next two activities are about testing algorithms and how they work.
- Help the class to connect their crocodile models to the computer as shown in the LEGO® Education WeDo manual.
- In pairs, have the class build each of the algorithms and execute the program to test it.
- The class must identify which of the algorithms works correctly and put a tick on their Resource sheet J.
- The class must also identify which of the algorithms do not work correctly and put a cross on their Resource sheet J.
- Check the students' answers to see who has correctly identified the working algorithm.

Which of the algorithms works correctly?

Why do the others not work?

Can you identify which program block is stopping the algorithm from working properly?

MAIN ACTIVITY 4

- Depending on how well the performed in the previous activity, the following activity could be done as a class or in pairs.
- Ask the class to pick one of the algorithms and attempt to add or replace a block that will make it work correctly. They are attempting to recreate the 'correct' algorithm on Resource sheet I as closely as possible.
- Pairs who finish debugging the algorithm must attach their hardware to the computer and execute the program to see if it now works.
- The class will either continue to do this until the algorithm works correctly, or move on to a new incorrect algorithm

Which program block needs to be replaced in the algorithm?

Is there a block missing?

How do you check if you have debugged the program properly?

Does the program execute properly now?

PLENARY

Have the class discuss what the algorithm did and how the program affected the hardware. Can they describe how the motor made the gears move the LEGO model? Can they describe how they think the sensor works?

Set them the challenge of explaining to an adult before the next lesson what their model did and how the motor and gears worked together.

Session 2.2
Resource sheet I

CORRECT ALGORITHM

PROGRAM BLOCKS WITH DEFINITIONS

This block will start the program

This block will open the jaws. The reverse arrow will close the jaws

This block will tell the motor how long to run for (using the time input block below)

This block will tell the program to wait. The motor will stop running at this point

This block tells the program to run when the sensor is triggered

This block tells the program how long something should run or wait for

Session 2.2
Resource sheet J

SPOT THE CORRECT ALGORITHM

SESSION 2.3

The Hungry Crocodile

LEGOeducation.co.uk

LEGO education

Session 2.3
Lesson overview

This lesson is designed to get the class to predict the behaviour of a simple program by discussing what they think an algorithm will do and applying logical reasoning. The lesson ends with a problem-solving task that requires them to make predictions for and debug another simple program.

LEARNING OUTCOMES

- Predict the behaviour of a simple program through logical reasoning
- Predict and debug a second simple program

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Provide reasons for their predictions of what they think a simple program will do
- Solve a problem using prediction and debugging skills

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a crocodile using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo Hungry Alligator building instructions and have pre-built the model in advance of the lesson.

ABOUT THE SOFTWARE

Following on from the previous lesson, the class will be using the LEGO® Education WeDo Software in this lesson to create and debug two simple programs.

It is recommended that you are familiar with adding and changing program blocks in the LEGO® Education WeDo Software.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 2.3
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, LEGO® Education WeDo Software, laptops/computers, and Resource sheet K and L.

Session 2.3
Lesson Plan

INTRODUCTION

Play some short prediction games, such as predicting the outcome of a flipped coin, rolling a 6 on a die. Discuss the language of prediction and having to justify your answer with a reason.

Predict the behaviour of a simple program through logical reasoning.

MAIN ACTIVITY 1

- In pairs, have the class build the crocodile model from the building instructions.
- In previous lessons you have helped the class to do this. This lesson is a good opportunity to see which pairs can build without assistance. You can focus your attention on those pairs that struggle with the build.
- Review what they remember about debugging a simple program from the previous lesson. Can they remember what has to happen to correct an algorithm in the software? Can they remember examples of how they did this?
- Question the class about debugging while they build.

What does debugging mean?

How do you debug a simple program?

Who has an example of what they changed or added to their algorithm in the previous lesson?

MAIN ACTIVITY 2 (45 MINUTES)

- Hand out Resource sheet K, which shows a series of similar algorithms. Discuss with the class what they observe using a similar questioning technique to the previous lesson.
- What do they notice is similar about each one? Which blocks do the algorithms all have in common? Which ones are different?
- Discuss as a class what you think any unfamiliar symbols on the blocks show and, therefore, what they think the block will do. Resource sheet L includes each new, unfamiliar block with a short description of its function.
- Ask the class to identify which algorithm will do each of the following:
  – Open and close the crocodile's jaws using the sensor.
  – Open and close the crocodile's jaws with a sound.
  – Open the crocodile's jaws and then snap shut.
- Remember that the class must predict which algorithm will make a program work in a certain way and try to justify their prediction using their knowledge of the program blocks used in the algorithm.

What is similar about each algorithm?

Which blocks do they have in common?

Which ones are different?

How do you think the different blocks will change the program?

What do you think each program will do?

Can you predict how it will affect the movement of the hardware?

Session 2.3
Lesson Plan

Solve a problem through prediction and debugging a simple program.

MAIN ACTIVITY 3 (45 MINUTES)

- Explain to the class that this activity is about testing algorithms.
- This activity follows on from the previous one, but begin by introducing the new learning objective above. Tell the class that they will apply the skills they have just been practising to solve a specific task.
- Tell the class that they need to pick the algorithm that they think will make the crocodile open it's jaws for the count of 10 and then close them for the count of 10 with a crunch sound (this is Sound 17, but do not tell them this yet).
- Connect their hardware to the software, build the identified algorithm and then execute the program.
- None of the algorithms will do exactly what they want it to, however, the algorithm below is closest.

- Students who pick the incorrect algorithm could be directed towards the correct starting example.
- In pairs or working in small groups for wider discussion, the class must decide which block(s) will need to be added or replaced to make the program execute properly. The algorithm below is one correct solution (also shown on Resource sheet L).

PLENARY

Have the class connect their hardware to the software and execute their debugged programs, observing as a class whether their changes worked.

Discuss how successful they were in predicting the closest algorithm and how they went about debugging their program. It does not matter at this stage if they made the program work properly, more that they were able to justify which algorithm they originally picked and reason why they made particular changes. Every student should feel like they succeeded, even if they only added or changed one block.

Which algorithm do you think is most likely to open the crocodile's jaws?

Can you spot if there are any particular programs with that algorithm?

What if it does not execute exactly as you want it to?

Which blocks might be causing the problem?

What would you change or add to the algorithm to make the program execute properly?

Session 2.3
Resource sheet K

PREDICT THE CORRECT ALGORITHM

Session 2.3
Resource sheet L

HUNGRY CROCODILE ALGORITHM SOLUTION

PROGRAM BLOCKS WITH DEFINITIONS

This block will add a sound effect to the program. Each number has a different sound

(Left click the number to increase the value, right click to decrease)

SESSION 3.1

Sports Stars

LEGO education

Session 3.1
Lesson overview

This lesson is designed to reintroduce the class to the basic elements of computing and programming. They will review how to create and execute simple programs, as well as being challenged to design a program with a specific goal.

You will ideally need to show plenty of sporting video clips and images.

LEARNING OUTCOMES

- Review how to create and execute simple programs
- Design a program with a specific goal

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain how they create and execute simple programs
- Describe what their simple program is doing/what it was designed to do

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build a kicking machine using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is important to point out the expectations of using the set with the class, for example, making sure that sets are put away just as they receive them.

ABOUT THE SOFTWARE

The use of the LEGO® Education WeDo Software to create basic programs is introduced in this lesson.

We recommend that you familiarise yourself with the software, in particular the various program blocks, so that you can encourage the class to explore them when creating their own programs.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 3.1
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, building instructions and Software, Resource sheet M and N, various sporting video clips and images, and laptops/computers.

Session 3.1
Lesson Plan

INTRODUCTION

Look at Resource sheet M with the class. Consider supplementing this with video clips or images of famous sporting moments. Talk about which sports they like most and who their sporting stars are.

Create and execute a simple program.

MAIN ACTIVITY 1

- Review basic programming and the LEGO® Education WeDo Software with the class. You could use any of the previous Resource sheets to stimulate dialogue about the elements of programming.
- Show the class the building instructions for the LEGO® Education WeDo Goal Kicker.
- Discuss briefly how precise the building instructions are and how there is little room for getting the build wrong, as they are so clear. This is something the class should be familiar with.
- In pairs, have the class build the Goal Kicker model from the building instructions.
- While they do this, ask the class to predict what they think the model will do. How will it use the sensor mounted on the front of the base plate?.

What do you remember about programming?

Can you give a definition for any of the following keywords: algorithm, program, hardware, software, execute, debug, motor, sensor?

What does the Goal Kicker model do?

How does it use the sensor?

MAIN ACTIVITY 2

- Hand out Resource sheet N, which shows an algorithm for programming the Goal Kicker. Discuss with the class what they recognise from the blocks in the algorithm.
- Review with the class how to connect the hardware to the software.
- In pairs, have the class create the algorithm on Resource sheet N and execute it using their Goal Kicker model.
- Ask the class to identify which blocks in the algorithm make the Goal Kicker move how it does. How is the sensor involved? What does the motor do?

Which of the program blocks are familiar?

What do you think it will make the model do?

Which blocks make the program execute in this way?

How is the sensor involved?

What does the motor do?

Session 3.1
Lesson Plan

Design a program with a specific goal.

MAIN ACTIVITY 3 (1 HOUR)

- Introduce the new learning outcome and define this next period as a task for which the class will determine the outcome.
- Explain to the class that they will need to design a new program, possibly by debugging or changing the algorithm from Resource sheet N, to achieve a specific goal.
- Goals could include changing the angle of the kick, changing how the program repeats, or making the kick more accurate to score in a particular goal/box/score area. Can they chip the ball?
- Discussion and decision time must be built into this lesson. This could be done as a whole class or in small groups. Ensure the class have realistic expectations of what their model and program can achieve.
- A table tennis ball is perfect for this activity.

PLENARY

Have some of the pairs show their model and explain how they designed their program to meet a specific goal. Ask the class to question them on what they did and how they did it. Discuss what the class have learnt from the group presentations.

Set them the challenge of explaining to an adult what they did with their Goal Kicker model and how they created a program to make it do something specific.

How can you define a specific goal for the hardware to achieve?

How will you design the program?

Could you use the suggested algorithm and modify it?

Session 3.1
Resource sheet M

PICTURES OF SPORTING ACTIVITIES FOR DISCUSSION

Key search terms:

David Seaman best save ever

Usain Bolt 100m world record

Jessica Ennis long jump, Heptathlon London 2012

Session 3.1
Resource sheet N

GOAL KICKER ALGORITHM

GOAL KEEPER ALGORITHM

SESSION 3.2

Sports Stars

LEGO education

Session 3.2
Lesson overview

This lesson is designed to introduce students to the process of using sequencing, selection and repetition when designing a program. Later in the lesson, the students will work in bigger groups – necessitating a higher level of classroom management.

During the lesson, you can refer back to the sporting images and video clips featured in the previous lesson.

LEARNING OUTCOMES

- Explore how to use sequencing, selection and repetition

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain how they have applied their knowledge of sequencing, selection and repetition in a program they have designed

ABOUT THE HARDWARE

This lesson includes an activity where the class are required to build Goal Kicker and Goal Keeper models using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo building instructions and have pre-built the Goal Keeper model, in advance of the lesson.

ABOUT THE SOFTWARE

Following on from the previous lesson, the class will be using the LEGO® Education WeDo Software in this lesson to begin exploring sequences, selection and repetition.

It is recommended that you are familiar with adding program blocks in the LEGO® Education WeDo Software and have pre-built the algorithm on Resource sheet N in the software in advance of the lesson.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 3.2
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, building instructions and Software, Resource sheet N and O, and laptops/computers.

Session 3.2
Lesson Plan

INTRODUCTION

The class will demonstrate and discuss how points are scored in various sports and how the opposing team try to stop them from scoring.

Explore the use of sequencing, selection and repetition in a program.

MAIN ACTIVITY 1 (45 MINUTES)

- Explain that, in pairs, the class will build the Goal Kicker model from the previous lesson as independently as possible.
- Explain that they will need to build the algorithm from Resource sheet N used in the previous lesson and connect the hardware to the software. They will then execute the program.
- You should try to keep a hands-off approach to this activity as much as possible, helping only those students who need help with the build or programming. This is an excellent opportunity to assess how well the class can apply their own knowledge and skills.

What do we mean by independent?

How can you work together to solve your own issues?

Who else could you ask before me?

How have you identified and solved the issues?

MAIN ACTIVITY 2 (45 MINUTES)

- Help the class so that they can follow the building instructions to build the Goal Keeper model in pairs.
- Connect the hardware to the software and hand out Resource sheet N if the class did not use it in the previous lesson.
- In pairs, have the class build the Goal Keeper algorithm from Resource sheet N and execute the program to test it.
- Discuss what the algorithm does. How is it different from the Goal Kicker program used in the previous activity?
- Identify which other blocks are different between the two algorithms.
- Also, introduce the idea of combining the Kicker and Keeper models and what the outcome might be. This will be explored in the next activity and during the third lesson.

How is the Goal Keeper program different to the Goal Kicker?

Which program blocks are different?

Why do the two algorithms need to be different?

How might you be able to combine the two models and algorithms?

Session 3.2
Lesson Plan

MAIN ACTIVITY 3

- Explain to the class that you will combine two pairs to form a group of four. It is important to establish that LEGO® elements must not be shared or swapped between the two sets.
- Explain how sequence, selection and repetition are used in programming to create a loop for hardware to continue to interact without stopping or needing to be reset.
- Show the algorithm on Resource sheet O (also shown below) and point out that blocks have been selected to create a sequence that repeats. Have the class point out which blocks they know. Can they predict what the unfamiliar blocks will do? How do they know a block has been selected to do a particular job?
- Explain that the yellow box around the algorithm below tells the program to repeat from the beginning, instead of stopping.

- Have the groups of four discuss what algorithm they would need to define in order to make the Goal Kicker and Goal Keeper interact together repeatedly. Will they be able to create one program that controls both models?

Which program blocks show that the algorithm will be repeated?

Which program blocks have been selected to do a particular job?

Are you familiar with any of these selected blocks?

PLENARY

Have the students share with the class their ideas for how the two models will interact. Can they explain what movements will need to be repeated? Which blocks would they select? Is there a specific sequence to their program yet or does this need to be explored further?

Set them the challenge of finding electronic devices or appliances around their house that work using specially selected algorithms that are designed to repeat without stopping or needing to be reset.

Appliances might include: computers, TVs, electric ovens, thermostats, washing machines, digital alarm clocks, smoke detectors, timer switches.

Session 3.2
Resource sheet 0

SEQUENCE, SELECTION AND REPETITION ALGORITHM

SESSION 3.3

Sports Stars

LEGO education

Session 3.3
Lesson overview

This lesson is designed to have the class create and debug a program with a specific goal, based on the discussion at the end of the previous lesson. At 1 hour long, the second activity allows the class time to break down the problem-solving aspect of the task into smaller parts, in order to find a solution. The class will work in their larger groups as determined during the previous lesson.

LEARNING OUTCOMES

- Design and debug a program with a specific goal
- Solve a problem by breaking it down into smaller parts

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, students must be able to:

- Explain how they have designed and debugged a program with a specific goal
- Solve a problem by breaking it down into smaller parts and solving each part

ABOUT THE HARDWARE

This lesson includes an activity that requires the class to build two models using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo Goal Kicker and Goal Keeper models from previous lessons.

ABOUT THE SOFTWARE

Following on from the previous lesson, the class will be using the LEGO® Education WeDo Software in this lesson to create and debug simple programs.

It is recommended that you are familiar with adding and changing program blocks in the LEGO® Education WeDo Software.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are three main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 3.3
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, LEGO® Education WeDo Software, laptops/computers and Resource sheet N, P, Q and R.

Session 3.3
Lesson Plan

INTRODUCTION

Present the class with one of the problems listed on Resource sheet P. Have them discuss how they would divide this problem into smaller parts to help find a solution.

Design and debug a program with a specific goal.

MAIN ACTIVITY 1

- Explain to the class that they will build the Goal Kicker and the Goal Keeper models in the same groups as the previous lesson.
- It is important to re-establish that LEGO® elements must not be shared or swapped between the two sets.
- Show the class the algorithms for the two models on Resource sheet N. Connect and execute the programs to make the two models work.

What do you want the two models to do?

How should they interact?

What might the implications be for the program?

Solve a problem through predicting and debugging a simple program.

MAIN ACTIVITY 2 (1 HOUR)

- Explain to the class that they have one hour to explore the following task.
- They will need to solve a problem: to design and debug a program with a specific goal. The problem is that the two models will need to interact, using two algorithms that combine into one executable program.
- There are many solutions for this task, so it is important that the class are allowed to explore the different program blocks. Part of the design and debug phase will be to experiment with different blocks.
- The students should decide on roles within the group, or specific parts they might work on, to develop their collaboration skills. Roles might include: designer, builder, programmer, tester, scribe, group leader.
- Guide the class towards the example solution programs on Resource sheet Q. In this solution, the two algorithms are executed by pressing the A and B keys on the keyboard respectively.
- Breaking for mini-plenaries to check-in with groups and having them explain what they have learnt so far offers assessment opportunities and guidance for other groups.

What have you done so far?

What program blocks have you explored?

How are the two models interacting?

How is everyone in your group involved in the designing and debugging process?

What is your specific role?

Given more time, what else would you like the program to do?

What are the most difficult parts of solving this problem?

Session 3.3
Lesson Plan

MAIN ACTIVITY 3

- Explain that after one hour, each group will present their models and programs to the rest of the class.
- Reinforce that a 'finished' product is not essential. What is more important is how the groups explains what they have learnt.
- In the form of a short presentation, groups might choose to explain:
  - What they wanted the models to do
  - Which blocks they selected
  - How they attempted to create a sequence
  - How they attempted to create a repeating program
  - How the group worked together
  - What roles they played within the group
  - How their program performed
  - What went well with their program
  - What they could/would improve next time
- Resource sheet R can be introduced as a way for the class to assess their own work during this lesson, for homework, or at a later time.

PLENARY

Have the class connect their hardware to the software and execute their debugged programs, observing as a class whether their changes worked.

Discuss how successful they were in predicting the closest algorithm and how they went about debugging the program. It does not matter at this stage if they made the program work properly, more that they were able to justify which algorithm they originally picked and reason why they made particular changes. Every student should feel like they succeeded, even if they only added or changed one block.

Did you solve the problem? If so, how?

What were the difficult parts of solving the problem?

How did you apply what you had learnt about programming?

What were you most successful at as a group?

What were you most successful at as an individual?

Session 3.3
Resource sheet Q

EXAMPLE SOLUTIONS

Pressing the A key on the keyboard will start the Goal Kicker (algorithm A). Once started, this algorithm will repeat every time the sensor detects a ball to kick, until stopped.

Pressing the B key on the keyboard will start the Goal Keeper (algorithm B). Once started, this algorithm will repeat until stopped.

For this program you can run the two algorithms through separate software, or connect the two motors to the same hub.

Session 3.3
Resource sheet R

SELF-ASSESSMENT SHEET

How did we solve the problem?

What were the difficult parts of solving the problem?

How did we apply what we have learnt about programming?

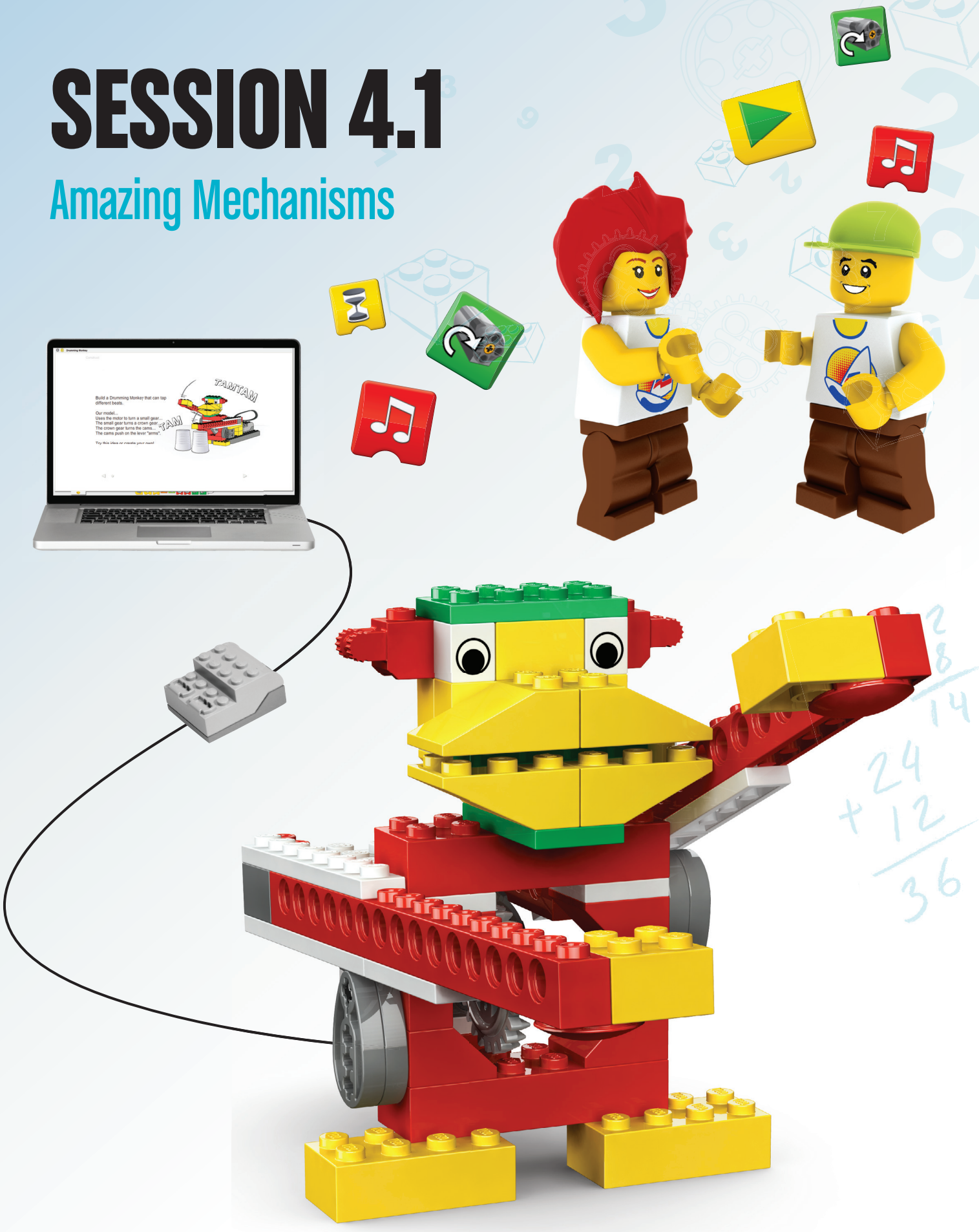
What were we most successful at as a group?

What was I most successful at as an individual?

What have I learnt to do better?

SESSION 4.1

Amazing Mechanisms



LEGOeducation.co.uk

LEGO education

Session 4.1
Lesson overview

This lesson is designed to reintroduce the class to designing and debugging simple programs. The class will then work with variables, such as power, inputs and directions. You will need to be familiar with the LEGO® Education WeDo Software.

For best results, you will also need to show examples of toys that move without an electronic/computerised input, such as spinning tops and toys with gears.

LEARNING OUTCOMES

- Review how to design and debug simple programs
- Explore working with variables

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, children must be able to:

- Explain how they debugged an 'incorrect' algorithm to execute properly
- Describe how different variables change the output of a program

ABOUT THE HARDWARE

This lesson includes an activity that requires the class to build a cam-driven toy using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is important to point out the expectations of using the set with the class, for example, making sure that sets are put away just as they receive them.

ABOUT THE SOFTWARE

The use of the LEGO® Education WeDo Software to design and debug programs is reviewed in this lesson.

It is recommended that you are familiar with the software, especially the variety of program blocks available in the software, so you can encourage the class to explore variables.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are two main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 4.1
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, building instructions and Software, Resource sheet S, various mechanical toys and laptops/computers.

Session 4.1
Lesson Plan

INTRODUCTION

Look at Resource sheet S and any non-electronic mechanical toys you can find (spinning tops, pull-back cars, toys with gears). Discuss how they work.

Review how to design and debug a simple program.

MAIN ACTIVITY 1 (1 HOUR)

- Review basic programming and the LEGO® Education WeDo Software with the class. You could refer to any of the previous resource sheets to stimulate dialogue about the elements of programming.
- Show the class the building instructions for the LEGO® Education WeDo Dancing Birds model.
- Discuss how precise the building instructions are and how there is little room for getting the build wrong because they are so clear. This is something the class should be familiar with.
- If you have already completed cams/mechanical toys as a Design Technology unit, discuss what the class observe about the Dancing Birds model.
- In pairs, have the class build the Dancing Birds model using the building instructions.
- Help the class connect the hardware to the software.
- Hand out Resource sheet T. Create an algorithm using these 'incorrect' instructions.
- Tell the class that the hardware should spin, and could also incorporate music or sound for the birds to dance to.
- Explain that the class will now need to debug the program to make sure it works properly. More confident pairs may wish to redesign it completely to work more efficiently.
- Guide the class towards the example solution program on Resource sheet T. This can be shared with the class at any point during the lesson. They will need to replace the motor blocks with the X symbol (which is actually switching the motor off) with a motor block with an arrow (which will make the motor run).
- Breaking for mini-plenaries to check-in with pairs and have them explain what they have learnt so far offers assessment opportunities and guidance for other groups.

How do the mechanical toys work?

What do gears, pulleys, cams and levers do?

What do you remember about programming?

Can you recall the definition of any of the following key words: algorithm, program, hardware, software, execute, debug, motor, sensor?

What did you notice was wrong about the algorithm?

What have you done so far?

What program blocks have you explored?

Have you been able to use sound?

Session 4.1
Lesson Plan

MAIN ACTIVITY 2 (1 HOUR)

- Introduce the new learning outcome and define this next period as a task for which the class will determine the outcome.
- Explain to the class that they will need to design a new program, possibly by debugging/changing the algorithm from Resource sheet T, to achieve a specific goal.
- Goals could include working with different variables, for example changing the motor power, adding random inputs, experimenting with how to execute the program in different ways, adding sounds, varying the speed and direction of the birds, and adding pauses into the program.
- There is no one desired outcome for this activity and it does not need to be defined at the beginning. However, the class must be prepared to feed back to the class what they have done and how they included different variables in their program.
- As with the first activity, breaking for mini-plenaries to check-in with pairs and have them explain what they have learnt so far offers assessment opportunities and guidance for other groups.

PLENARY

Have some of the pairs show their model and explain how they designed the program to include some of the explored variables. Ask the class to question them on what they did and how they did it. Discuss what the class have learnt from the group's presentation.

Set them the challenge of explaining to an adult how mechanical toys work and how they can be controlled by a computer program.

Which variables have you worked with?

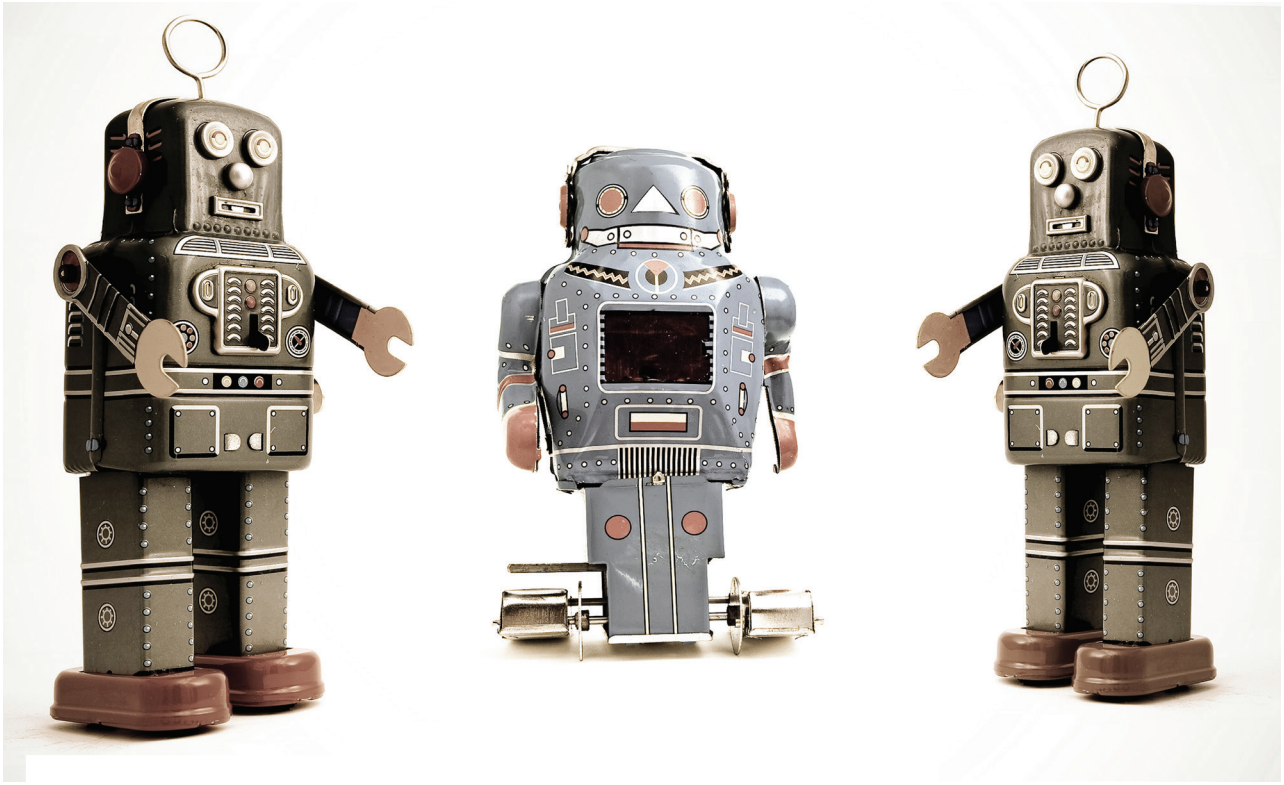
What have you observed has happened to the birds as you apply the changes?

Does the program work better with the changes? If so, why?

What else could you change, if you had more time?

Session 4.1
Resource sheet S

PICTURES OF MECHANICAL TOYS FOR DISCUSSION



Session 4.1
Resource sheet T

INCORRECT DANCING BIRD ALGORITHM

DANCING BIRD ALGORITHM SOLUTION

SESSION 4.2

Amazing Mechanisms

LEGO education

Session 4.2
Lesson overview

This lesson is designed to introduce the class to exploring and understanding how variables can affect the outcome of a program. It prepares the class for an open-ended design and test activity in the next lesson.

During the lesson, you can refer back to the mechanical toys and pictures featured in the previous lesson.

LEARNING OUTCOMES

- Design and debug a program
- Explore how to combine programs, using variables, with a specific goal

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, children must be able to:

- Explain the process of designing and debugging a program
- Justify the variables they used to meet the specific goal

ABOUT THE HARDWARE

This lesson includes an activity that requires the class to build the Drumming Monkey and Dancing Birds models using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo building instructions. You may want to have pre-built the two models in advance of the lesson.

ABOUT THE SOFTWARE

The LEGO® Education WeDo Software is used again this lesson, as a continuation of the previous lesson.

It is recommended that you are familiar with the LEGO® Education WeDo Software and have pre-built the algorithm on Resource sheet V in the software, in advance of the lesson.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are two main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 4.2
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, building instructions and Software, Resource sheet S, T, U and V, and laptops/computers.

Session 4.2
Lesson Plan

INTRODUCTION

Review what the class know about mechanical toys, using Resource sheet S and mechanical toys you collected in the previous lesson. Are the class familiar with any mechanical toys that work and interact together, for example battling spinning tops?

Design and debug a program.

MAIN ACTIVITY 1 (1 HOUR)

- Show the class the building instructions for the LEGO® Education WeDo Drumming Monkey model. Discuss how they think the monkey will work and what kind of program blocks the algorithm might contain.
- In pairs, have the class build the Drumming Monkey model from the building instructions.
- Help the class connect the hardware to the software.
- Hand out Resource sheet U. Create the very simple, two-block algorithm using the example instructions.
- Explain that the class will now need to explore the program blocks, and then design a new algorithm that makes the monkey play different beats.
- Guide less-confident students to the example algorithm on Resource sheet U. Have them explain the differences and what they would need to change to create this new algorithm.
- The class should be able to predict how changes to the algorithm will affect the way the monkey moves. Encourage them to use as much key vocabulary as possible during explanations.
- Breaking for mini-plenaries to check-in with pairs and have them explain what they have learnt so far offers assessment opportunities and guidance for other groups.
- By now, students should be confident sharing experiences in small groups or with the whole class. If time allows before the end of the lesson, it would be useful for students to discuss what they have done so far with at least one other pair.

What do you think the monkey will do?

What program blocks will be included in the algorithm?

Can you justify your predictions for which program blocks will be included in the algorithm?

What have you changed in the algorithm?

How has it affected the monkey?

What needs to change to make the new algorithm on Resource sheet U?

Session 4.2
Lesson Plan

Explore how to combine programs, using variables, with a specific goal.

MAIN ACTIVITY 2 (1 HOUR)

- Explain to the class that you will combine two pairs to form a group of four. It is important to establish that LEGO® elements must not be shared or swapped between the two sets.
- Review how sequence, selection, repetition and other variables are used in programming to create a loop for hardware to continue to interact without stopping or needing to be reset.
- Show the algorithms on Resource sheet V, and point out that blocks have been selected to create a sequence that repeats. Have the class point out how they know the blocks repeat.
- Have one pair build the Dancing Birds model from the building instructions, while the other pair work with the Drumming Monkey model. This pair can begin to explore and review the algorithms they have been given.
- As a group, have the two pairs design and debug a program that combines algorithms for both models. Less confident groups can use the algorithms on Resource sheet V as a starting point, but all group should consider how to use variables from the previous lesson.
- As before, breaking for mini-plenaries to check-in with the small groups and have them explain what they have learnt so far offers assessment opportunities and guidance for other groups.
- By now, all students should be encouraged to share their experiences with the whole class. If time allows before the end of this lesson, it would be useful for students to discuss what they have done so far with at least one other group.

How can you apply your knowledge of sequence, selection, repetition and variables?

Which blocks show that the algorithm will be repeated?

Which blocks show variables that you can explore?

How might you get the two models to interact in new ways?

How would the algorithms need to be designed to allow the two models to interact?

Can the monkey create the beat for the birds to dance to?

PLENARY

Have the class share their experiences of how much trickier the task becomes with the new variables to consider. How does this create issues when working as a group? How did they overcome problems they encountered? Can they identify students in the class who are particularly good at programming?

Session 4.2
Resource sheet U

EXAMPLE DRUMMING MONKEY ALGORITHM

EXAMPLE DRUMMING MONKEY ALGORITHM TO PLAY A DIFFERENT BEAT

Session 4.2
Resource sheet V

EXAMPLE COMBINED PROGRAM

Pressing the A key on the keyboard will start the Dancing Birds.

Pressing the B key on the keyboard will start the Drumming Monkey.

For this program you can run the two algorithms through separate software, or connect the two motors to the same hub.

SESSION 4.3

Amazing Mechanisms

LEGO education

Session 4.3
Lesson overview

This lesson is designed to have the class to solve an open-ended problem, including documenting a written program. It is the culmination of the units of work in this book and requires the class to work independently in a pair to design and debug a program using the Dancing Birds and Drumming Monkey hardware.

LEARNING OUTCOMES

- Design a program and detect and correct errors in it
- Solve a problem, including documentation of a written program

These are shown at the relevant points in the lesson plan in a coloured box.

By the end of the lesson, children must be able to:

- Explain how and why they have designed a program in a particular way
- Create a written record of the program they designed

ABOUT THE HARDWARE

This lesson includes an activity that requires the class to build two models using LEGO® bricks. All the bricks they need can be found in a LEGO® Education WeDo Construction Set.

It is suggested that you are familiar with the LEGO® Education WeDo Amazing Mechanisms models from previous lessons.

ABOUT THE SOFTWARE

Following on from the previous lesson, the class will be using the LEGO® Education WeDo Software in this lesson to create and debug simple programs.

It is recommended that you are familiar with adding and changing program blocks in the LEGO® Education WeDo Software.

ABOUT THE LESSON PLAN

Each lesson starts with a short introduction and ends with a plenary, both designed to be 10 minutes long. There are two main activities to carry out for this lesson.

Assessment for learning opportunities are provided through questions posed throughout the lesson plan in a series of coloured callouts.

These are easy to adapt and supplement to fit your needs. They offer guidance for using mini-plenaries throughout the lesson.

All resource sheets and algorithm examples referenced in the lesson plan are provided within the resources section for each lesson.

Session 4.3
Lesson overview

VOCABULARY

Underlined words in the lesson plans are regarded as key computing vocabulary that the class must learn. They are only underlined the first time they appear in each lesson.

All key computing vocabulary is listed in a glossary at the end of the book, along with a child-friendly definition to share with the class.

RESOURCES

This lesson requires: LEGO® Education WeDo Construction Sets, LEGO® Education WeDo Software, laptops/computers, and Resource sheet W, X and Y.

Session 4.3
Lesson Plan

INTRODUCTION

Review Resource sheets W, X and Y with the class. Tell them that they will guide the class through the documentation and self-assessment process.

Design a program and detect and correct errors in it.

Solve a problem, including documentation of a written program.

MAIN ACTIVITY 1 (1.5 HOURS)

- Explain to the class that in pairs, they will be designing a program using either or both the Dancing Birds and Drumming Monkey.
- Explain to the class that they have 90 minutes to explore the following task. This time should include using Resource sheets W, X and Y to document their process.
- The students could decide that hardware will need to interact, in which case they will need to team up with another pair in the class.
- There are many possible ideas and solutions for this task, so it is important that the class are allowed to explore the different program blocks, depending on what they want the hardware to do. Part of the design and debug phase will be to experiment with different blocks.
- The class should have free rein to create what they like, including modifying either or both of the models. The LEGO® Education WeDo Teacher's Guide has more information on how the models can be modified, including a greater variety of ways to use the program blocks.
- Breaking for mini-plenaries to check-in with pairs or groups and have them explain what they have learnt so far offers assessment opportunities and guidance for other groups.
- At each step, the class will need to test the algorithms they create using the LEGO® Education WeDo Software, detecting any errors and correcting them through the debug routine they have practised with in previous units.
- If necessary, you may help the students decide what they are going to create. Ideas include:
  - Make the Dancing Birds dance differently
  - Use the Drumming Monkey to create the beat for the Dancing Birds
  - Recreate either model using a different mechanism (pulley and belt, lever, cam, gears, spinning top, sensor activated).

What do you want the hardware to do?

How did you decide they should interact?

What might the implications be for the program?

What have you done so far?

What program blocks have you explored?

How is the hardware interacting?

How did you assign roles within your pair or group?

Given more time, what else would you like the program to do?

What are the most difficult areas of solving this problem?

Have you learnt anything new about programs and programming?

Session 4.3
Lesson Plan

Explore how to combine programs, using variables, with a specific goal.

MAIN ACTIVITY 2 (30 MINUTES)

- Explain that after 90 minutes, each group will present their models and programs to the rest of the class.
- Reinforce that a 'finished' product is not essential. What is more important is how the pair or group will explain what they have learnt.
- In the form of a short presentation, using Resource sheet W, X and Y for support, pairs or groups might choose to explain:
  - What they wanted the hardware to do
  - How they planned for the hardware to interact
  - Whether they modified the builds or built their own
  - Which programming blocks they selected
  - How they detected errors in the program
  - How they debugged and corrected those errors
  - How the pair or group worked together
  - What roles they played within the pair or group
  - How their program turned out
  - How they were able to document their program in written form
  - Which Resource sheets, if any, they used
  - What went well with their program
  - What they would improve next time

PLENARY

Have the pairs or groups connect their hardware to the software and execute their debugged programs one at a time, observing as a class whether their changes worked.

Discuss how successful they were in predicting the closest algorithm and how they went about debugging their program. It does not matter at this stage if they made the program work properly, more that they were able to justify which algorithm they originally picked and reason why they made particular changes. Every student should feel like they succeeded, even if they only added or changed one block.

What did you want the hardware to do?

How did you plan for the two models to interact?

What were the difficult parts of solving the problem?

How did you apply what you had learnt about programming?

What were you most successful at as a pair or group?

What were you most successful at as an individual?

How would you improve the process next time?

Session 4.3
Resource sheet W

OUR CREATION – DOCUMENTATION SHEET 1

Our design

What we designed _____

What we designed it _____

Some of the program blocks we used

We used this block because _____

We used this block because _____

We used this block because _____

Our creation did/didn't turn out as we planned, because _____

OUR PROGRAMS – DOCUMENTATION SHEET 2

Session 4.3
Resource sheet Y

SELF-ASSESSMENT PROMPT SHEET

How did we solve the problem? How successful was the outcome?

What were the difficult parts of solving the problem? Why were they difficult?

How did we work together to apply what we have learnt about programming?

What were we most successful at as a pair or group? Why?

What was I most successful at as an individual? Why?

What have we learnt to improve our teamwork or task outcome next time?

Glossary

Algorithm: an algorithm is a set of instructions a computer uses to get something done. We use algorithms in our daily lives, such as when getting dressed or making a cup of tea, to ensure we do everything in the right order.

Block: program blocks are used in the LEGO® Education WeDo Software to build an algorithm. It uses blocks with symbols instead of making you write out the code in words.

Cam: a rotating part in a machine designed to create motion by pushing something up and down, such as in the Dancing Birds model.

Debug/debugging: the process a computer programmer goes through to find mistakes and correct them. We often debug our written work when we look for spelling mistakes and missing punctuation, so that it can be read correctly.

Digital devices: we are surrounded by digital devices, such as computers, game consoles, tablets, mobile phones, televisions, cameras, remote controls and MP3 players.

Execute: to carry out something we have been asked to do until it is completed. Computers execute programs.

Gears: a gear is a toothed wheel that rotates and makes something else work. You can find gear wheels in your watch, on your bike and in a car.

Hardware: hardware is the correct term for a piece of machinery or equipment. Your LEGO models are hardware. You can tell the difference between hardware and software because hardware is something physical you can touch.

Instructions: a set of instructions give you orders or directions to help you get something done. You will notice that LEGO building instructions do not use any words, only pictures.

Motor: a motor is a powered object that makes other things move. The motor in a car uses petrol to supply power to the wheels to make it move. LEGO® Education WeDo motors use electricity to make a plastic rod rotate.

Program: a computer program tells the computer, step by step, exactly what you want it to do. You can choose how you want the computer to do something by using algorithms. A computer cannot work without a program.

Pulley and belt: a pulley is driven by a belt. The pulley is a wheel with a groove in it where the belt rests. The LEGO® Education WeDo belt is like a small rubber band, which connects to a part of the model that is rotating, transferring the rotation to a different part of the model.

Repetition: a computer programmer can use specific program blocks or written code that tells an algorithm to keep repeating. This means it will keep doing the same thing until it is told to stop, rather than just doing it once.

Selection: a computer programmer must select the best program blocks for a particular algorithm, depending on what they want it to do. There is often more than one way to create a program, so selecting the best block for the job is important to make the program efficient.

Sensor: a sensor detects or measures something (such as distance) and converts it into a digital signal. You will be using the LEGO® Education WeDo distance sensor. This will detect when an object (such as a finger or LEGO brick) is placed in front of it. When the sensor is used in an algorithm it will tell the program when to execute a particular task, such as turning on the motor. The LEGO® Education WeDo Construction set also includes a tilt sensor.

Sequence: it is important for a programmer to put program blocks or written code in the correct order. Sequencing is used to create a structure where one action in the algorithm leads to the next action.

Software: software is the correct term for a program that a computer uses to work. Where hardware is a hard, physical model you can touch, software runs inside the computer and can only be seen on the computer screen. Computer games, spreadsheets and web browsers are examples of software you are probably familiar with.

Variables: as in science, a variable is an element, feature or factor that can be changed or varied in some way. When a variable is changed the program will be affected, such as by making a motor run faster or playing a sound louder.