

Module 1 - Activity 4:

The Pink Robot...

Overview

In this lesson students will complete their first debugging activity. They will run the code, identify the problem and work to debug the code.

Computing PoS Reference

- Write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by breaking them into smaller parts.
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.

Learning Objective

- To decompose the route and debug the algorithm.

Success Criteria

All: I can debug the code and test as I debug.

Most: I can decompose the route, debug the code and test as I debug.

Some: I can decompose the route, debug the code using clues from the algorithm and test as I debug.

Key Words

debug, decompose, angle, direction, motion, block, algorithm, sequence, code, event.

Computer Science Concepts

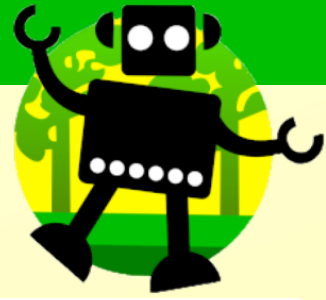
- Algorithm design
- Debugging
- Event
- Motion

Cross Curricular Concepts

- Mathematics: turning, angles, degrees, distance.

Introduction

Share the learning objective and success criteria. Play the introduction and explain to the students that they will be debugging (fixing) code. Ask them if they have had to debug their code during previous activities.



Activity 4: The Pink Robot

EXPLORE: Run the first program.

Q: What is wrong with the code? Which line of code do we need to change and how?

OFFLINE ACTIVITY: Use the classroom to set up a similar scenario, with students acting as beetles to test distance and turns.

DISCUSS: Ask the children how they would debug the code and demonstrate how to change the values using one idea. Run the program.

Q: Did it work? What do we do if the code is incorrect? Are there any clues in the algorithm which will help us work out the correct distance and angle?

CHALLENGE: Ask the students to work through the debugging challenges in pairs, keeping track of their attempts on the Record Sheet.

EVALUATE: Discuss with students how they completed the challenges.

Q: Which challenges were the most difficult and why? How did you find the line of code that needed changing? Were there any clues in the algorithm that helped you decide on the correct distance and angle?

SEN Support

Use the Support Sheet and/or Record Sheet to check the code and record ideas. If necessary, allow the students to walk through the coding until they find the incorrect block.

Resources

- Record sheet
- Support sheet
- Answer sheet

Extension Activity

Write an algorithm to draw a spiral then test it on Scratch or ask a friend to reproduce the algorithm on paper (forward 70 = forward 70mm). Think about the pattern you will need to code.

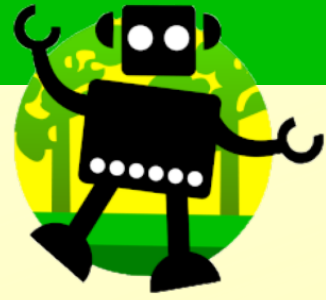
Possible Key Questions For Assessment

Can you give two top tips for debugging?

What skills do you need to debug an algorithm?

Why is it important to be able to decompose the algorithm?

Which clues in the code helped you to work out the correct distance or angle?



Activity 4: Example Answers

when clicked
move 200 steps
turn 90 degrees
move 100 steps

Well done!

when clicked
move 50 steps
turn 90 degrees
move 150 steps
turn 90 degrees
move 200 steps

Well done!

when clicked
move 50 steps
turn 90 degrees
move 200 steps
turn 90 degrees
move 200 steps

Well done!

when clicked
move 100 steps
turn 90 degrees
move 100 steps

Well done!

when clicked
move 50 steps
turn 90 degrees
move 100 steps
turn 90 degrees
move 150 steps

Well done!

when clicked
move 150 steps
turn 90 degrees
move 150 steps

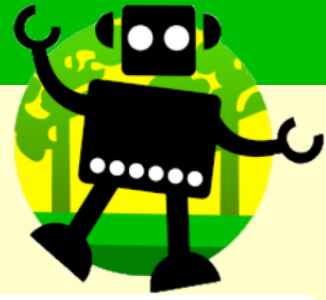
Well done!

when clicked
move 50 steps
turn 90 degrees
move 100 steps
turn 90 degrees
move 150 steps

Well done!

when clicked
move 50 steps
turn 90 degrees
move 50 steps
turn 90 degrees
move 100 steps
turn 90 degrees
move 100 steps

Well done!




Activity 4: Support Sheet 1


Which coding block needs debugging?

Look through the coding block-by-block. Tick each line that would work, until you find the block of code that is incorrect.

Make notes on the sheet. Now go back to the game and try your idea to see if you are correct. If not, have another go!

when  clicked

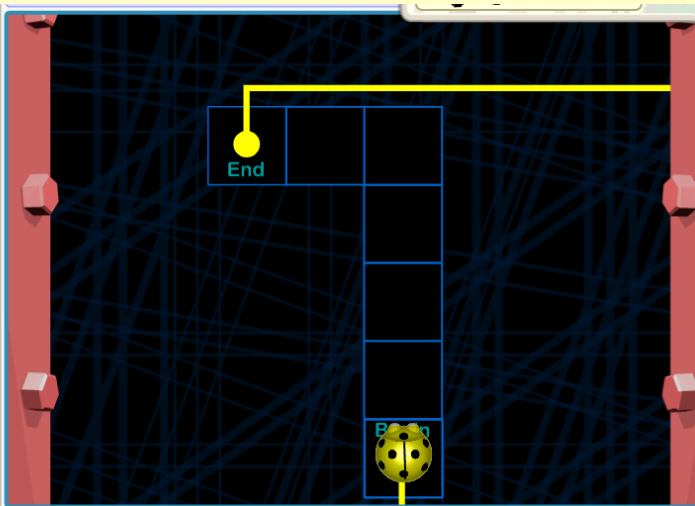
move 200 steps

turn  90 degrees


move 150 steps

This is the next connection.
But the beetle's program has a bug in it.


Run the beetle






when  clicked

move 50 steps

turn  90 degrees

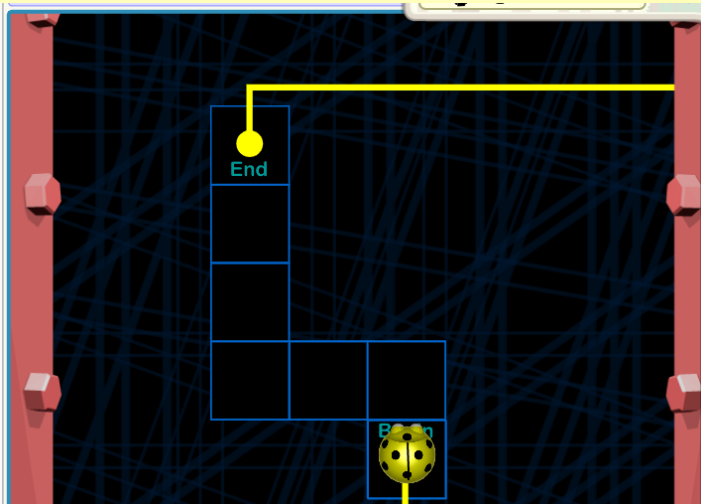
move 50 steps

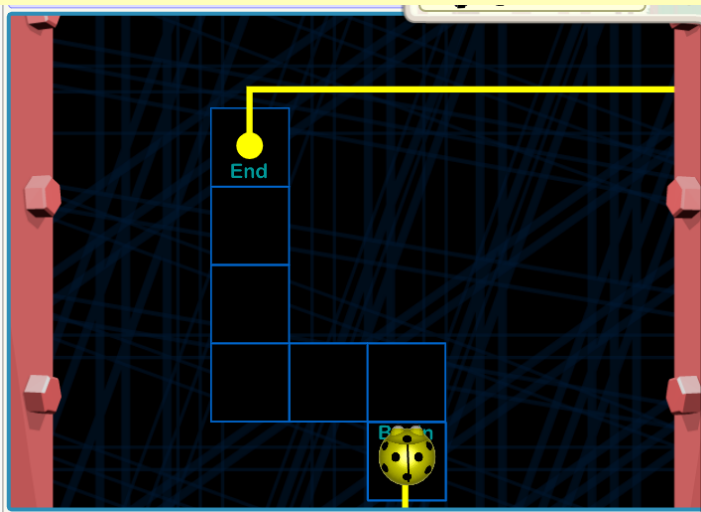
turn  90 degrees

move 150 steps

This is the next connection.
But the beetle's program has a bug in it.

Run the beetle







Activity 4: Support Sheet 2

when clicked

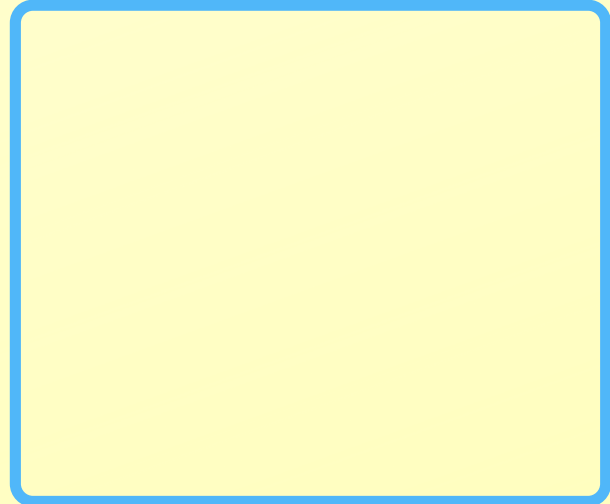
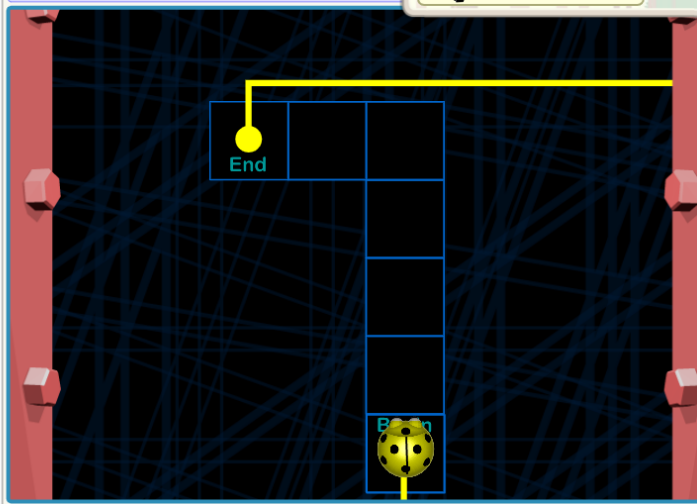
move 200 steps

turn 90 degrees

move 150 steps

This is the next connection.
But the beetle's program has a bug in it.

Run the beetle



when clicked

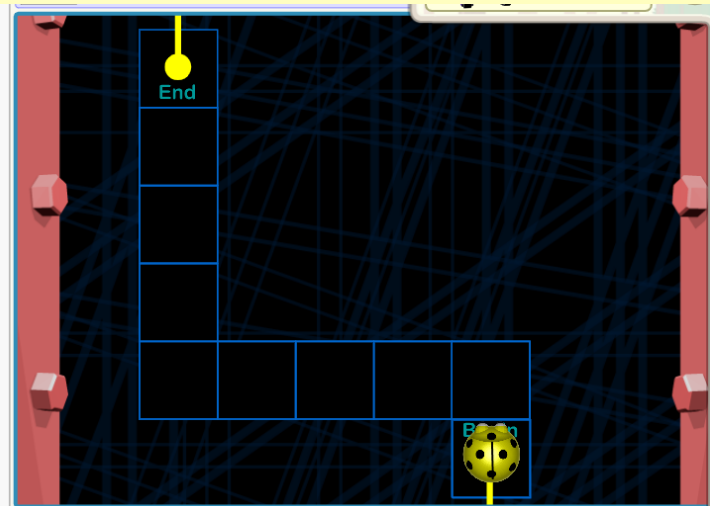
move 50 steps

turn 90 degrees

move 200 steps

turn 90 degrees

move 100 steps



when clicked

move 50 steps

turn 90 degrees

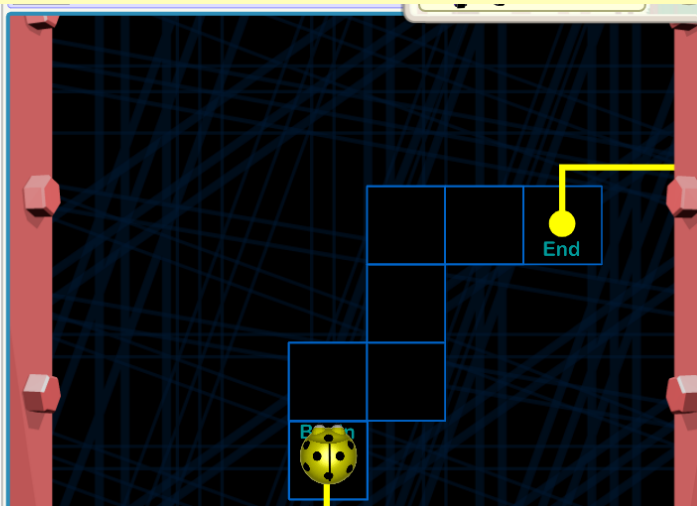
move 50 steps

turn 90 degrees

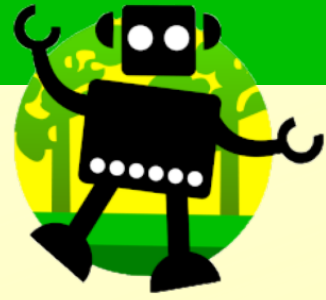
move 150 steps

turn 90 degrees

move 100 steps



Activity 4: Record Sheet



when  clicked