



## Year 3 – Programming B – Events and actions in programs

### Unit introduction

This unit explores the links between events and actions, while consolidating prior learning relating to sequencing. Learners begin by moving a sprite in four directions (up, down, left, and right). They then explore movement within the context of a maze, using design to choose an appropriately sized sprite. This unit also introduces programming extensions, through the use of **Pen** blocks. Learners are given the opportunity to draw lines with sprites and change the size and colour of lines. The unit concludes with learners designing and coding their own maze-tracing program.

There are two Year 3 programming units:

- Programming A – Sequencing sounds
- Programming B – Events and actions in programs

This is unit B, which should be delivered after unit A.

### Overview of sessions

Session	Brief overview	Learning objectives
1 Moving a sprite	In this session, learners will investigate how characters can be moved using 'events'. They will analyse and improve an existing project, and then apply	To explain how a sprite moves in an existing project

	what they have learned to their own projects. They will then extend their learning to control multiple sprites in the same project.	<ul style="list-style-type: none"> <li>• I can explain the relationship between an event and an action</li> <li>• I can choose which keys to use for actions and explain my choices</li> <li>• I can identify a way to improve a program</li> </ul>
2 Maze movement	In this session, learners will program a sprite to move in four directions: up, down, left, and right. They will begin by choosing a sprite and sizing it to fit in with a given background. Learners will then create the code to move the sprite in one direction before duplicating and modifying it to move in all four directions. Finally, they will consider how their project could be extended to prove that their sprite has successfully navigated a maze.	<p>To create a program to move a sprite in four directions</p> <ul style="list-style-type: none"> <li>• I can choose a character for my project</li> <li>• I can choose a suitable size for a character in a maze</li> <li>• I can program movement</li> </ul>
3 Drawing lines	This session will introduce learners to extension blocks in Scratch using the <b>Pen</b> extension. Learners will use the pen down block to draw lines, building on the movement they created for their sprite in Session 2. Learners will then decide how to set up their project every time it is run.	<p>To adapt a program to a new context</p> <ul style="list-style-type: none"> <li>• I can use a programming extension</li> <li>• I can consider the real world when making design choices</li> <li>• I can choose blocks to set up my program</li> </ul>
4 Adding features	In this session, learners will be given the opportunity to use additional <b>Pen</b> blocks. They will predict the functions of new blocks and experiment with	To develop my program by adding features

	<p>them, before designing features to add to their own projects. Finally, they will add these features to their projects and test their effectiveness.</p>	<ul style="list-style-type: none"> <li>• I can identify additional features (from a given set of blocks)</li> <li>• I can choose suitable keys to turn on additional features</li> <li>• I can build more sequences of commands to make my design work</li> </ul>
5 Debugging movement	<p>This session explores the process of debugging, specifically looking at how to identify and fix errors in a program. Learners will review an existing project against a given design and identify bugs within it. They will then correct the errors, gaining independence as they do so. Learners will also develop their projects by considering which new setup blocks to use.</p>	<p>To identify and fix bugs in a program</p> <ul style="list-style-type: none"> <li>• I can test a program against a given design</li> <li>• I can match a piece of code to an outcome</li> <li>• I can modify a program using a design</li> </ul>
6 Making a project	<p>In this session, learners will design and create their own projects. Using a template (which can be blank or partially completed), learners will complete projects to move a sprite around a maze, with the option to leave a pen trail showing where the sprite has moved. Ideally, projects will include setup blocks to position the sprite at the start of the maze and clear any lines already on the screen.</p>	<p>To design and create a maze-based challenge</p> <ul style="list-style-type: none"> <li>• I can make design choices and justify them</li> <li>• I can implement my design</li> <li>• I can evaluate my project</li> </ul>

## Progression

This unit assumes that learners will have some prior experience of programming. The key stage 1 National Centre for Computing Education units focus on floor robots and ScratchJr, however experience of other languages or environments may also be useful. The Year 3 — Programming A unit introduces the Scratch programming environment and the concept of sequences.

See the learning graph for this unit for more information about progression.

## Curriculum links

### National curriculum links

- 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 some simple algorithms work and to detect and correct errors in algorithms and programs
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

## Assessment

### **Formative assessment**

Assessment opportunities are detailed in each session plan. The learning objective and success criteria are introduced in the slide deck at the beginning of each session and then reviewed at the end. Learners are invited to assess how well they feel they have met the learning objective using thumbs up, thumbs sideways, or thumbs down.

### **Summative assessment**

See the assessment questions and solutions for this unit.

We recommend the use of teacher and learner accounts in Scratch to help with assessment throughout this unit. For guidance on setting up teacher accounts, visit [scratch.mit.edu/educators/faq](https://scratch.mit.edu/educators/faq). A teacher account enables you to manage learners' accounts and organise projects into studios. If you are unable to use teacher and learner accounts, work can be saved offline to local devices.

### Subject knowledge

This unit focuses on the links between 'events' and 'actions' in programming, while also developing learners' understanding of sequencing. It highlights that events cause actions, and that the order of those actions can have an impact on the outcome of a program. This unit also further develops learners' understanding of design in programming, using the approach outlined below.

When programming, there are four levels that help to describe the stages of a project, known as levels of abstraction. Research suggests that this structure can support learners in understanding how to create a program and how it works.

- Task — this is what is needed
- Design — this is what it should do
- Code — this is how it is done
- Running the code — this is what it does

Spending time at the Task and Design levels before engaging in code writing aids learners in assessing the 'do-ability' of their programs and reduces a learner's cognitive load during programming.

Learners will move between the different levels throughout the unit. This is highlighted within each session plan.

Enhance your subject knowledge to teach this unit through the following training opportunities:

### Online training courses

- [Raspberry Pi Foundation online training courses](#)

### Face-to-face courses

- [National Centre for Computing Education face-to-face training courses](#)

Resources are updated regularly — please check that you are using the latest version.

This resource is licensed under the Open Government Licence, version 3. For more information on this licence, see [nccce.io/ogl](https://nccce.io/ogl).