Years 3-4

Programming and algorithms and KLA examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Specific DT focus** | **Subject area** | **Short focus** | **Summary** | **Title** |
| Pathways | English | Clear, precise communication | In pairs, explore giving and following a sequence of steps and decisions to build a LEGO® toy. | Take a [LEGO® building challenge](#LEGOBUILDING) |
| Decision making (branching) | Create a storyboard to plan a story where the reader is provided with a number of decisions that lead to alternative endings. | Plan a ['choose your own adventure' story](#PLANCYOA) |
| Mathematics | Decision making (branching) | Create a flowchart to represent a sequence of (branching) steps and decisions needed to solve a mathematical problem. | Have [fun with flowcharts](#FunWithFlowcharts) |
| Choices | English | Implementing a digital solution using visual programming | Based on your storyboard for a 'choose your own adventure' story, use a visual programming language to implement a digital solution. |  |
| Create a multimodal game board where the player is provided with a number of decisions. Using Scratch and Makey Makey, add multimodal elements to the story. These are activated using an Ozobot. | [Create a game board that uses an Ozobot](#Ozobot) |
| Sequencing instructions to complete a task | Plan and create a computer program to demonstrate grammar or spelling rules, eg changing nouns from singular to plural; adding ‘ing’. |  |
| HAAS: History | Implementing a digital solution, using visual programming | Design and create a simple game/quiz to demonstrate convict crimes and punishments. | Design a quiz –[Convicts: crime and punishment](#Convict) |
| HAAS: Geography | Implementing a digital solution, using visual programming | Create a computer program for learning a traditional Aboriginal or Torres Strait Islander language. | [Create a language -learning program](#languagelearning) |
| Science | Implementing a digital solution, using visual programming | Design and create a simple quiz to explore the difference between living and non-living things. |  |
| Mathematics | Implementing a digital solution, using visual programming | Modify an existing program or create a program to design a geometric shape or design using Pencil code or similar application. |  |
| Programming a robotic device to follow a path | Create a maze or route for a programmable robot to travel. Estimate and calculate angles and distances. |  |
| Working together | HPE | Collaboration | Collaborate to decide the rules for a new game. Use a flowchart to explain the consequences of unfair play. |  |

****

**TITLE:** Take a LEGO® building challenge

***SUB HEADING:***Giving and following instructions

**Summary text** In pairs, explore giving and following a sequence of steps and decisions to build a LEGO® toy.

Year Level: Years 3–4



# Suggested steps

1. Working in pairs, students complete the construction of a LEGO® toy. One student interprets the printed instructions into step-by-step verbal instructions for the other student to follow to build the toy. The student following instructions does this without speaking or seeing the printed instructions.

   To create engagement, students could pick which object they want to build from a list of downloadable instructions on Lego.com. The link is: <https://www.lego.com/en-us/themes/classic/building-instructions> -- The greater variety of instructions may also support those who have difficulty with fine motor control, and remembering instructions, as well as those with vision impairment.

1. Both students then compare the completed toy against the printed instructions, to see whether these instructions were effectively communicated (verbally) and followed.

**Variations**

1. If using the 3 in 1 LEGO® toy kit, then repeat the LEGO® toy-building activity, constructing a different version of the three possible toys, but this time include a time limit. This will encourage students to understand the importance of breaking tasks down, of seeing patterns in the way they describe things so they are easily understood, and of abstracting concepts to complete a set task.
2. Students build their own character or model from LEGO® or similar building blocks. Each student then instructs their partner to use the relevant blocks to build a replica of their creation. (Ensure sufficient blocks are available to make the replica.)

**Extension Task:** Have students write down their instructions for their partner to build the object without verbal instructions.

# Discussion

* Discuss commonly used instructional language. Some examples: pick up, join, fix, press; directional language (over, on top of, below, next to, beside); and specific technical language (the red 4x2 block).
* Talk about decisions: At what stages (if any) did students have to make a choice about what to join or what order of steps to follow? What instructions were used? (Look for any examples of if-then statements.)
* Discuss approaches to minimising/clarifying instructions: clear and precise instructions in correct order, summarising the instructions in sub tasks (# now we are adding headlights) ‘Pick up the clear, single, cylinder block and place on the end of the 4x1 white block’.
* Connect algorithms to programming: Different groups came up with different algorithms to solve the same problem and code can be written (in different ways) to implement the algorithm.

# Why is this relevant?

Computers operate by following a list of instructions, called a program, which has been written to carry out a particular task. Programs are written in languages that have been specially designed, with a limited set of instructions, to tell computers what to do. Some languages are more suitable for some purposes than others.

This task provides students with the opportunity to create and follow an algorithm to complete a particular task, focusing on clear and precise language provided in the correct order.

**Note**: this task has limited opportunity to incorporate choices (though that is an integral part of designing algorithms at this level (years 3–4).

# Assessment

**Self-assessment**

Students consider the algorithm designed to instruct their partner and the one they followed.

They write three things they learned about providing step-by-step instructions. What suggestions can they make to improve their clarity?

They use an emoji to describe how they worked with their partner.

# Australian Curriculum alignment

* Technologies / Digital Technologies / Years 3 and 4 / Digital Technologies Processes and Production Skills / Define simple problems and describe and follow a sequence of steps (algorithms) needed to solve them [(ACTDIP010)](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACTDIK002)
* English / Year 3 / Literacy / Interacting with others / Use interaction skills, including active listening behaviours and communicate in a clear, coherent manner using a variety of everyday and learned vocabulary and appropriate tone, pace, pitch and volume [(ACELY1792)](http://www.australiancurriculum.edu.au/Curriculum/ContentDescription/ACELY1792)