

# Space Adventures Computing Unit 2

## Lesson 4 – Lunar Landing

### Curriculum Mapping (Computing KS2)

- ◆ 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

### Learning Objective

Code a simple game to land Tazz's lunar module on the Moon.

### Prior Learning

Good knowledge of most KS2 coding techniques, familiarity with Scratch.

### Introduction

Explain that a landing on the moon must be done as gently as possible. Spaceships slow down by firing rockets in the opposite direction, but this uses fuel so must be done sparingly. Show pupils the **U2L4 introduction.mp4 video**. Using the prompts in the video, ask pupils to explain the rules that make the lunar lander move. (This can be done orally or written in rough, and is known as the algorithm).

### Main Activity

Pupils use Scratch to create their own lunar landing game. Pupils will use variables to keep a track of fuel used and the speed of the lunar module/spaceship.

Show the class the **U2L4 demonstration.mp4 video** or how to access it on their own computers.

Hand out the **U2L4 step by step.pdf** guide or show pupils how to access it on their computers.

(Opening a second tab in the browser will allow pupils to switch between the help guide/video and their own work).

### Extension Activity

Show pupils the activities on the **U2L4 going further.pdf** document. These include experimenting with aspects of the code and adding a score variable judging how good a landing was.

### Plenary

Discuss how this game simulates gravity, by making the spaceship move down, and accelerate.

Ask them to think of games that use acceleration/gravity in this kind of way (eg Flappy Bird). Ask them to plan a game of their own using this idea.

### Notes

Although this is not a very long program it does use some complex calculations to make things move around. Rather than just moving the spaceship down the same amount each loop, the speed variable increases simulating gravity. This makes the spaceship accelerate.

Pressing the up arrow button simulates the effect of a rocket thruster acting the opposite way and slowing the spaceship down. The if then code block uses an and operator to test for two things.

Firstly it checks to see if the up arrow is pressed, secondly to see if there is any fuel left. Only if both these things are true then the rocket thruster is fired.