**Computing fundamentals  
Lesson 2: Computational thinking 2** 

**Introduction**

In this lesson students deepen and extend their learning of computational thinking. They consider how they use the concepts covered in the last lesson in other subjects before designing a prototype of a computerised paper aeroplane to develop their understanding that using a computer to help solve a problem is an important step in computational thinking.

**You will need:**

Lesson plan, lesson guide and worksheets, large sheets of paper, various crafting materials (e.g. card, pens, glue, sticky tape).

**Learning objectives**

* To develop a deeper understanding of computational thinking concepts
* To understand the steps in computational thinking
* To develop and present a prototype and algorithm for a computerised paper aeroplane

**Lesson summary** approx 60 minutes

1. Computational thinking recap (5 minutes)
2. ‘Computational’ thinking (10 minutes)
3. Designing better paper aeroplanes with computers (10 minutes)
4. Creating a prototype (20 minutes)
5. Presenting prototypes (10 minutes)
6. Review & wrap up (5 minutes)

**1. Introduction: Computational Thinking recap (5 minutes)**

* Recap last lesson’s learning using **slides 3 to 8**.

**2. ‘Computational’ Thinking (10 minutes)**

* Highlight to students that the skills they have been using can be applied in many different fields and subjects of study.
* Have a quick challenge quiz to get teams to come up with one way they use each concepts in different subjects (**slide 9**) and briefly discuss examples (**slide 10**).
* Invite students to share how computational thinking is different from these general skills they use in other subjects and remind them that computational thinking is looking at a problem in a way that a computer can help us to solve it (**slide 11**).

**3. Designing better paper aeroplanes with computers (10 minutes)**

* Explain that they are going to use computational thinking to consider how computers could help to create a giant aeroplane that is ‘better’ (**slide 12**).
* Discuss what ‘Better’ could mean. E.g. it flies for a longer time or further, it can be directed or is more controllable - they can choose their own interpretation.
* Give out large pieces of paper to teams and invite them to brainstorm and share their ideas. Encourage students to be creative using the examples on slide 12 if you wish as a starting point.

**4. Creating a prototype (20 minutes)**

* Give out large pieces of paper and some basic crafting materials to students and explain that you would like them to create a paper prototype of their idea to present to the class (**slides 13 & 14**). Encourage creativity in how they create and present their prototype and algorithm.

**5. Presenting prototypes (10 minutes)**

* Ask each team to present their prototype and algorithm to the class, prompting them to consider how they used their computational thinking skills and inviting peer feedback.

**6. Wrap up (5 minutes)**

* Review the learning objectives if you wish on **slide 15** and invite students to create an ‘exit ticket’ for the lesson using the questions on **slide 16**.

**Extension ideas:**

* You could ask students to write a one page explanation of what computational thinking is and how they used it in the creation of their prototype.
* Teams could create questions (increasing in difficulty) to create a class Computational Thinking quiz.

**Differentiation**

**Support:**

* Students can focus on one aspect of computational thinking in the first activity; e.g. just thinking about where they use instructions (algorithms).
* They may need additional support and examples for coming up with ideas in the prototyping task, though there are no right and wrong ideas, so encourage creativity to build confidence.

**Stretch & challenge:**

* Students can be asked to think of more challenging examples in the first activity; e.g. further examples of abstraction.
* Encourage students to strive for more detail in their algorithm and prototype.
* Challenge students to consider and discuss potential problems with their prototype and how they could be approached.

**Opportunities for assessment:**

* Informal observation and assessment of students’ responses during team activities
* Observations made during the prototype creation challenge.
* More formal assessment of team’s prototype and individual responses to the exit ticket if wished.