**Introduction to cryptography**

**Lesson 2: Caesar cipher algorithms**

**Introduction**

In this ‘unplugged’ lesson students create their own Caesar cipher, firstly using card before writing, testing and debugging algorithms to create a Caesar cipher to encrypt and decrypt messages.

**Time:** approx. 60 minutes

**Learning objectives**

* To create and use a paper-based Caesar cipher to encrypt and decrypt messages
* To use logical reasoning to write algorithms to encrypt and decrypt messages using a Caesar cipher
* To test and debug algorithms effectively

**Materials needed:** Lesson plan, lesson guide, Caesar cipher worksheet, card, scissors, paper fasteners, A3 paper and pens for pairs/small groups.

**Lesson summary**

1. Introduction: recapping cryptography (5 minutes)
2. Creating card Caesar ciphers (20 minutes)
3. Caesar cipher algorithms: encryption (15 minutes)
4. Caesar cipher algorithms: decryption (10 minutes)
5. Review & wrap up (10 minutes)

**Introduction: Recapping cryptography (5 minutes)**

* Ask students to think/pair/share what they can recall about cryptography and ciphers from the previous lesson (**slide 2**), highlighting you will be focusing on Caesar ciphers in this lesson.
* Share the learning objectives on **slide 3** if you wish.

**Creating card Caesar ciphers (20 minutes)**

* Give out the **Caesar cipher worksheet** to individuals or pairs of students (**slide 4**) and go through the instructions to check their understanding.
* Ask students to collect card, scissors and paper fasteners and to create their ciphers.
* Once they have created the ciphers, students can complete task B on their sheet.
* Discuss their learning as a class, inviting students to share their answers to the ‘challenge’ question and highlighting examples of using logical reasoning (**slide 5**).

**Caesar cipher algorithms: encryption (15 minutes)**

* Explain to students they are going to write an algorithm giving for encrypting a message using a Caesar cipher and ask pairs or small groups to collect A3 paper and pens to write their algorithm.
* If you wish, you can discuss as a class to give some direction (there is a simple example on **slide 6**).
* Once students have written their first version, ask them to write a message at the bottom of their sheet and swap their sheet with another pair or group.
* Ask them to test each other’s algorithms by trying to encrypt the message using the algorithm’s instructions and to feed back any problems found in testing for the other pair/group to debug.
* Discuss as a class, so all have a working algorithm for encryption and ask students to collect a new sheet of A3 paper and encrypt a message using their algorithm at the top.

**Caesar cipher algorithms: decryption (10 minutes)**

* Ask students to write an algorithm to decrypt the message they have written using a Caesar cipher. This should be easier for them as they have already written the encryption algorithm, though you can use the simple example on **slide 7** to help if needed).
* Once they have their algorithm, again ask them to swap with another pair or group who should test it using the algorithm only to try to decrypt the message and feedback any problems to be debugged.
* Again, discuss as a class to address any misconceptions.

**Review & wrap up (10 minutes)**

* Use the questions on **slide 8** to prompt review discussion in students’ pairs or groups around their learning in this lesson.
* Discuss responses as a class, exploring students answers to question 5 in particular, rewarding any credible attempts generously even if the language or context is not quite right (e.g. lookup tables, variables, arrays, lists).
* Review students’ learning against the learning objectives on **slide 9** if you wish.

**Differentiation**

**Support:**

* Students may benefit from supportive pairings during the algorithm writing task, could work in a small group with adult support to sequence a set of pre-printed instructions or have a starter algorithm from which to continue.
* Ensure they have a short message to decrypt/encrypt and encourage any sign of increased understanding and confidence when writing algorithms and using iteration and selection.

**Stretch & challenge:**

* Students can be challenged to create a more detailed algorithm (e.g. one that allows a user to decide the number of letters to ‘shift’).
* Students can work independently or in pairs to explore using their algorithm to try to write a program in familiar software.

**Opportunities for assessment:**

* Informal observation of students’ during Caesar cipher and algorithm writing activities.
* Informal assessment during class, group and individual questioning.
* More formal assessment of algorithms if wishes.