Code Breakers : Dpef Csfblfst

Introduction

How many of us, when we were children, passed messages around the classroom to friends? How many of us had these messages read by our peers, or much to our embarrassment the teacher intercepted the message and read it (sometimes even out loud)? When I've asked pupils in my class these questions, I've found that more hands go up to the second question than the first. When I ask the same class of pupils who would want to be able to encrypt their messages to save any future embarrassment… every hand goes up!

Introducing the topic of code breaking and encryption to learners in this way creates an exciting hook and real life context for learners to engage with this fascinating topic. It shows how pupils can, to quote from the national curriculum, "use computational thinking and creativity to understand and change the world." Encryption has played an important role throughout history (I give some examples below) and continues to be an important part of our lives. Every time we make a phone call, send an email or purchase something on the world wide web, these tasks depend on confidentiality and security. All organisations are required by law to look after and keep secure the data they store, and encryption plays a part.

Lesson Ideas

There are several cipher methods we can use with pupils to encrypt and decrypt information in the classroom. I've described some of them in the "A menu of ciphers" section near the end of this chapter. You can use any of these ciphers for either of the "Using ciphers" or "Cracking ciphers" activities, though your students might find the later ciphers easier to understand after they've used the earlier ones.

Learning Objectives

At the end of this chapter, you should be able to:

- understand the importance of cryptography throughout history and in modern technology
- explain that the process of encryption and decryption are algorithms
- encrypt and decrypt messages using a range of simple ciphers

Links to teacher standards and National Curriculum

Teacher standards

1c: set goals that stretch and challenge pupils of all backgrounds, abilities and dispositions.

3a: have a secure knowledge of the relevant subject(s) and curriculum areas, foster and maintain pupils' interest in the subject, and address misunderstandings.

4b: promote a love of learning and children's intellectual curiosity.

4c: reflect systematically on the effectiveness of lessons and approaches to teaching.

5a: know when and how to differentiate appropriately, using approaches which enable pupils to be taught effectively.

6a: know and understand how to assess the relevant subject and curriculum areas, including statutory assessment requirements.

6b: make use of formative and summative assessment to secure pupils' progress.

6d: give pupils regular feedback, both orally and through accurate marking, and encourage pupils to respond to the feedback.

Programme of study

2.2 use sequence, selection, and repetition in programs; work with variables and various forms of input and output

2.3 use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

2.4 understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration

2.7 use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Need to know: essential subject knowledge for teachers and pupils

In cryptography, a cipher is an algorithm for performing encryption or decryption. Like any algorithm, it has a series of well-defined steps that can be followed as a procedure.

Encryption is a process of transforming meaningful data, which cryptographers call "plaintext", into indecipherable code, known as "ciphertext." Decryption is the process of turning the ciphertext back into the plaintext so that it can be understood by the intended viewer. To be able to decrypt the ciphertext, the viewer must know the cipher method and have the key used to be able to view the plaintext.

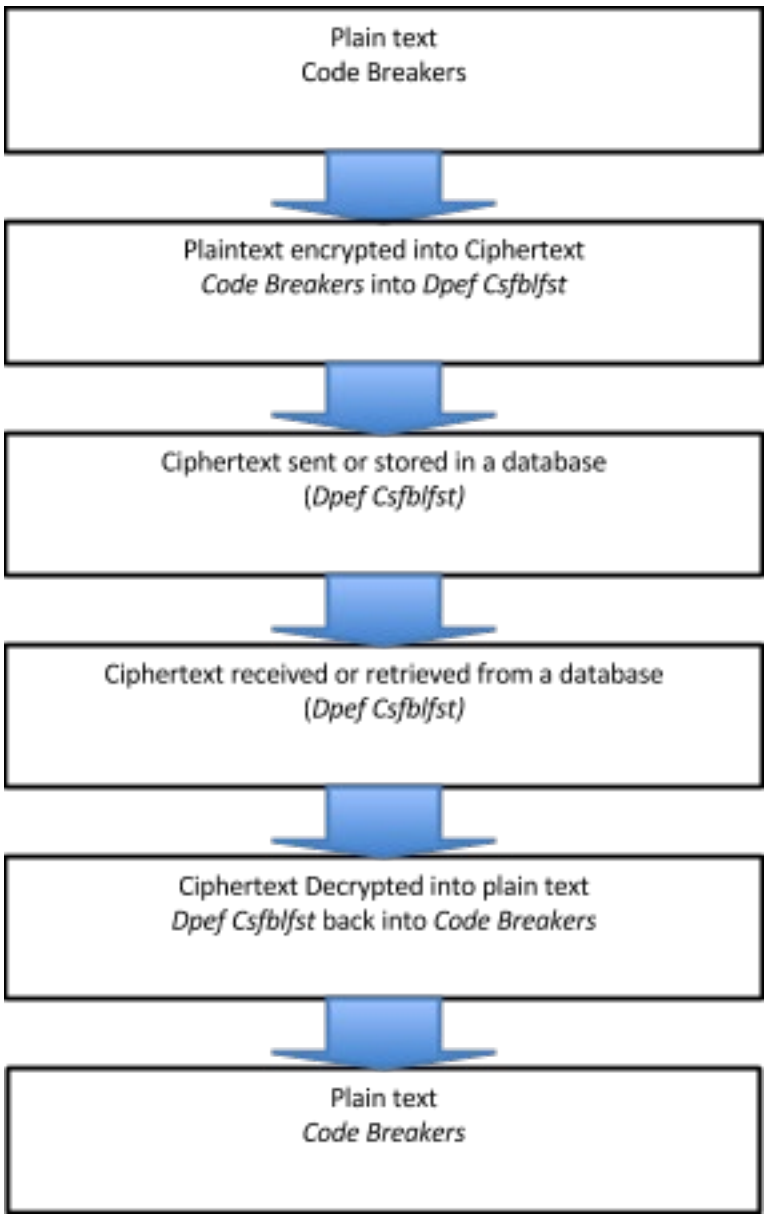


Figure 1: Enciphering and deciphering

Most cipher algorithms are well-known so the strength of a cipher depends on how hard it is for an eavesdropper to find the right key. Good ciphers have lots of keys and don't provide clues to them in the ciphertext.

While we use the terms "plaintext" and "ciphertext." ciphers can and do encrypt all sorts of data, including text, numbers, sounds and images. This brings out an important point about how information is stored in a computer as just ones and zeros (binary). As all data looks the same at this low level, the same ciphers can be used to encrypt and decrypt the data, whatever its interpretation at a higher level.

There are many methods of encrypting information, some simple and straightforward, others more complex. In this chapter, we will look at a few simple but effective examples to use in the classroom with pupils, and look at how to break codes.

Cross curriculum links:

Literacy (writing messages, understanding the alphabet), Numeracy (calculations in ciphers), History (historical uses of ciphers), PHSE (using ciphers for online safety), and D&T (making cipher devices such as the Caesar cipher wheel).

Computational thinking

The activities in this chapter develop a number of computational thinking aspects.

- Algorithmic thinking
    - Formulating instructions to be followed in a given order (sequence).
    - Grouping and naming a collection of instructions that do a well-defined task to make a new instruction (subroutines, procedures, functions, methods).
    - Creating algorithmic descriptions of real world processes so as to better understand them
- Abstraction
    - Reducing complexity by removing unnecessary detail.
    - Choosing a way to represent an artefact, to allow it to be manipulated in useful ways.
    - Hiding complexity in data, for example by using data structures.
- Decompostion
    - Breaking down artefacts into constituent parts to make them easier to work with.
- Evaluation
    - Assessing that an artefact is fit for purpose
    - Assessing whether an artefact does the right thing (functional correctness).
    - Assessing whether the performance of an artefact is good enough (utility: effectiveness and efficiency).
    - Comparing the performance of artefacts that do the same thing.
    - Assessing whether an artefact is easy for people to use (usability).
    - Assessing whether an artefact gives an appropriately positive experience when used (user experience).
    - Stepping through processes or algorithms/code step-by-step to work out what they do (dry run/tracing).
- Generalisation
    - Identifying patterns and commonalities in artefacts.
    - Adapting solutions, or parts of solutions, so they apply to a whole class of similar problems.

Activity 1: Using ciphers

You are a secret agent who needs to send and receive classified messages to a follow-spy (a classmate). You need to investigate and carefully choose a cipher method to encrypt and

decrypt your messages. Before you can send and receive your own classified messages you must return to spy training school.

Overview

There are a variety of ways of teaching how each algorithm for the ciphers work. My preferred method is to get pupils to break down the algorithms into smaller parts to make them easier to work with. Pupils do this by working their way through each algorithm (step-by-step) to work out what they do and then ask them to represent that as a sequence of instructions, using a standard notation, such as a flow diagram. They can then use these diagrams when attempting to crack the codes in Activity 2.

This same activity structure can be used with any of the ciphers described in the "A menu of ciphers" section below, or any other cipher you may decide to use.

Age range

Beginning KS2 onwards. Caesar and pigpen ciphers might be usable by KS1 children

Learning outcomes

After this activity, students should be able to do and achieve these:

- I understand what is meant by the terms ‘cryptography’, ‘encryption’, ‘decryptions’, ‘plaintext’ and ‘ciphertext’.
- I have a simple understanding of how encryption works and how it keeps personal and private information secure
- I can encrypt and decrypt messages using a simple cipher

Need to know

Cryptography is about keeping information secure so that even if it falls into the wrong hands it can’t be understood. A cypher is a method for turning information from a form that people can read into one that people can't (called encrypting), and reversing the process (decrypting) when the right person wants to read it.

Key words and questions

Cipher, plaintext, ciphertext, algorithm, key, password.

Activities

Time	Teacher activity	Student activity	Resources
10 minutes	Lead a discussion about secrets and private communication. Who would want to keep secrets, and why? Who would want to find out someone's secrets? Emphasise that everyone has secrets and private information, such as a teacher's first name, or a student's address.	Pupil should brainstorm some things that might be secret but still need to be communicated or stored. Secret messages between spies, information people want to keep from criminals, and so on.	

25 minutes	Give an example of a simple cipher, such as a Caesar cipher with a one-position shift. Show how to encipher and decipher messages with this cipher.	Pupils given a Caesar cipher wheel together with a word in ciphertext. Pupils should work out the position shift used e.g. A become S. Pupils should then encrypt a message using a Caesar cipher with a position shift of their choice.	Pupils could make and use a cipher wheel. See the resources below.
25 minutes	If there is time, explain how a range of ciphers work to pupils.	Pupils should encrypt and decrypt a number of simple messages using a range of ciphers.	For resources, refer to the Menu of ciphers below.

If pupils have seen and used a range of ciphers, they should evaluate the ciphers. Remember that good ciphers have lots of keys and don't provide clues to them in the ciphertext.

For each of the algorithms (ciphers methods) should think about:

- Does the algorithm do the right thing?
- How easy is the algorithm (cipher method) for people to follow?
- Is the algorithm (cipher method) fit for purpose? If not, why not?

Variations, differentiation, and extensions

The choice of encryption method is up to you. You should choose one (or more) that is appropriate for your learners. You should use simpler ciphers with younger and less able pupils.

Some older and more able pupils will naturally want to investigate the ideas of using two ciphers, one after the other, to encrypt and decrypt their messages. This is possible and should be encouraged because it is only a matter of stepping through two algorithms, after the other.

Success criteria and assessment

The evaluation criteria are directly related to the learning outcomes:

- Are pupils able to explain how encryption keeps personal and private information secure?
- Are pupils able to successfully perform the algorithm to encrypt plain-text into cipher-text and vice versa to produce a different representation of a word or message?
- Are pupils able to evaluate and compare a range of ciphers they use?

However, you should be aware of the range of computational thinking skills that can be developed by using ciphers and you may want to modify how you lead this activity to emphasise different skills.

Activity 2: Cracking ciphers

The term “hacker” is a popular way to describe somebody who illegally tries to break the cipher method to access the plaintext for which it was not intended they see. To develop pupils' computational thinking skills and associated attributes, it can be fun for pupils try hacking an encrypted word.

Age range

End of KS2

Learning outcomes

After this activity, students should be able to do and achieve these:

- I know what frequency analysis is and how it speeds up the code breaking process of cracking a substitution cipher.
- I understand that personal and private information need be kept safe and secure
- I understand the need for and use complex passwords and keep them secure
- I can persevere and try attempt to solve a problem with different methods

Need to know

Computational thinking is a framework and not a recipe. It is a framework to encourage pupils to ask good questions. So it is important to encourage the pupils to ask appropriate questions about the cipher method.

I give some approaches to breaking simple ciphers below. These approaches are not foolproof algorithms: applying one approach will not always allow someone to read an enciphered message. Therefore the cracking activity also develops the perseverance approach of computational thinking, encouraging pupils to keep trying to decipher messages even if their first attempt fails.

Key words

Hacker, transposition, Substitution, high-frequency, analysis

Activity

Begin by encrypting some words using one of the cipher methods pupils are familiar with. Give the pupils the word (either on individual sheets or using a board to show the whole class and challenge them to decipher it.

You can provide suggestions on how to break ciphers by giving pupils the information below on cracking ciphers. Even better would be demonstrate breaking one piece of ciphertext in front of the pupils.

You may want to spend anywhere between 120 – 180 minutes on this activity depending upon the age and ability of your pupils. It is impossible to allocate a length of time to each activity because this will depend upon the pupils understanding of the concepts being taught and their problem solving capabilities being applied.

When pupils have cracked the code, repeat the task but this time increase the single word to a short sentence and again telling them which cipher method you have used so that the pupils can consider the variables associated with that given method. Try using a different cipher method, this will help you assess the pupils understanding other cipher methods whilst reinforcing their computational thinking skills.

After one or two successful cracks, ask the pupils how they could work together to solve the task more quickly. How would they approach the task? What can they apply from the previous hacking tasks to this one?

Pupils should conclude that they should work as a team to reduce the amount of time it takes to crack the code, with some members tying some keys until they find one that works. This emphasises how tasks can be **decomposed** into smaller jobs, and how several tasks can be performed in parallel to improve overall performance.

Conclude this activity by leading a discussion with pupils. In this discussion it is really important that pupils apply what they have learnt about hacking of encrypted messages to their own computer behaviours, that is, understanding the importance of using complex passwords to keep their personal and private information secure from hackers. As part of this plenary ask pupils to consider:

- Do you trust your computer to keep things secret?
- Are there things that need to be kept secret or private?
- What things would you want encrypted?

Cracking Substitution ciphers

One approach to cracking a cipher is to try every key and decipher the entire message for each key. This would be a very slow and tedious method code breaking.

Instead code breakers will often turn to frequency analysis to help them identify a pattern in the ciphertext. This is because certain letters and combination of letters occurring with varying frequency. They are useful when a substitution cipher method has been applied to the plaintext.

The three most frequent letters in normal English are E then T and O. The title of this chapter contains the ciphertext “Dpef Csfblfst”. Therefore, if you know that the cipher method was a substitution cipher, and the letter F appears three times in the ciphertext, it is likely that 'F' replaces one of these letters. If pupils know the message used a Caesar cipher, they can turn their cipher wheel to their guess, setting E to equal F, then use that to decipher a fragment of the message. If the deciphered message looks sensible, they can decipher the rest. If not, they can use a different setting and hence a different trial key, such as seeing if T is enciphered to F.

The same technique could be applied to the polybius square ciphertext "13 34 14 15 12 42 15 11 25 15 42 43". The only difference when applying this technique is that instead of substituting the characters that appeared most in the ciphertext, pupils would substitute the high frequency number pair, that is, grid references.

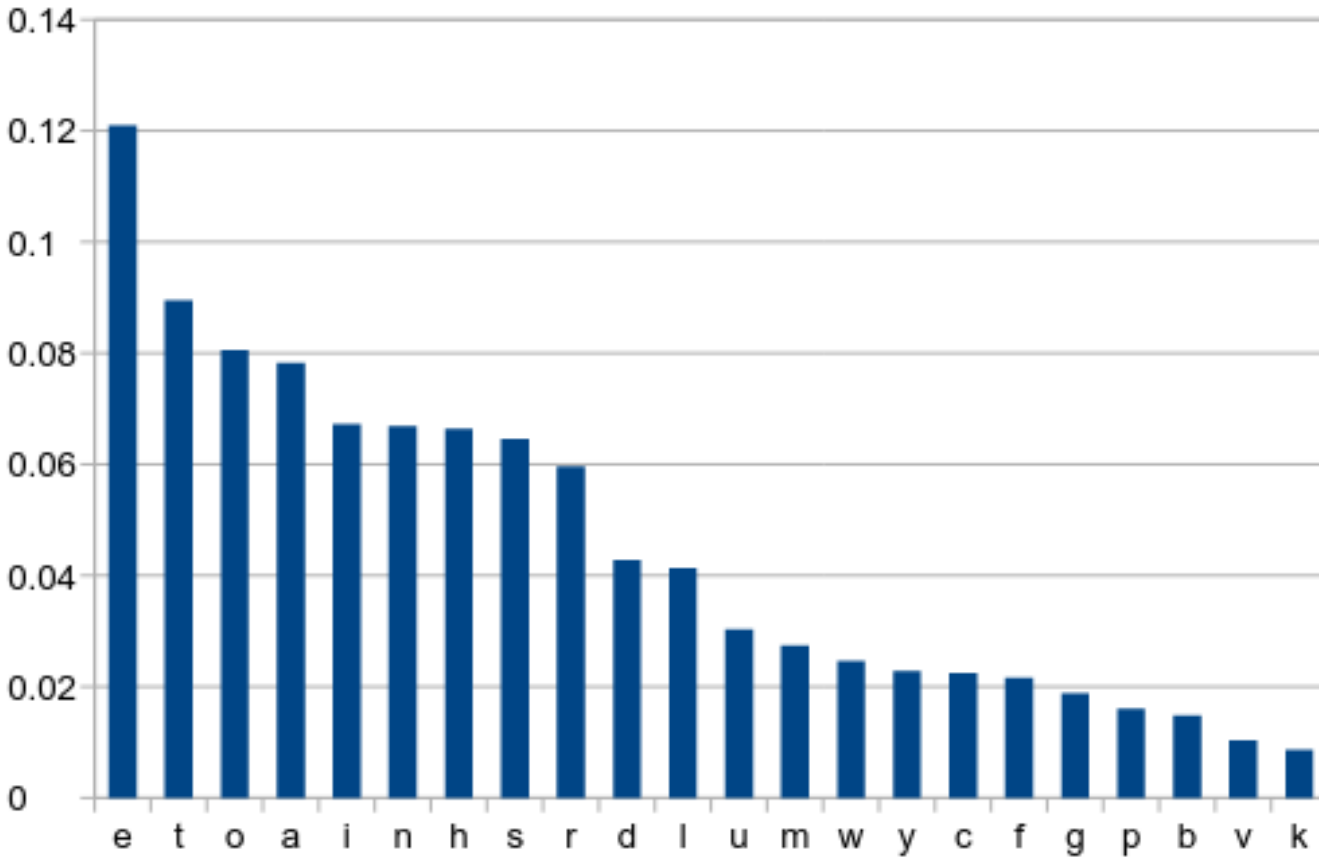


Figure (*): Frequencies of letter occurrences in English.

Cracking transposition ciphers

If you use a transposition cipher then it is important for pupils to think about what possibilities were available to you as the person encrypting the plaintext? Did you use a railfence method, and if so, how many rows could you have used given the length of the word? Or could it have been a route method, and again what are the likely grid dimensions?

Pupils will probably not think to collaborate with one another to complete the task. After a few minutes stop the pupils, again talk through what possibilities were available to you when encrypting the data. Through key questioning, encourage pupils to work as a team and divide the possible methods and associated variables between them to reduce the length of time it could take to crack the code.

Extension and differentiation

Whilst pupils develop their hacking skills, it is important to tell the pupils which encryption method you used otherwise it is too challenging for even the most able students.

Any easy way to adjust the difficulty in your choice of word. Words with double letters, or phrases with one- or two-letter words are easier to crack, as the features of the message give people a place to start. Short messages and/or words with unusual letter combinations are harder (words like “rhythm” are difficult).

When the pupils have undertaken and have cracked the code for a range of cipher methods, you can encrypt a short message in one of the cipher methods studied but this time you shouldn’t tell them which method you used.

Success criteria and assessment

This activity promotes both the **concepts** of computational thinking and some of its **approaches**. Many of the concepts used and developed in this activity are the same as for Activity 1. However, cracking ciphers requires tenacity, allowing students do develop **perseverance** as they try out different guesses for the cipher keys, along with **debugging** and **tinkering** as they progressively refine their guesses.

Activity 3: Encoding for transmission

Overview

Codes don't have to be used to conceal messages. They can also make it easier to transmit messages from place to place. Codes like semaphore or Morse code allow us to send messages across distances further than we can shout. Semaphore uses vision, while Morse code was invented for when telegraphs could only send a binary signal (the electrical switch was closed or open).

The Polybius cipher can be a good illustration of how changing the format of a message can make it easier to transmit. The Polybius square converts a message that uses 26 symbols in different combinations into one that only uses five symbols (typically the digits 1 to 5).

Age range

End of KS1 or beginning KS2

Learning outcomes

After this activity, students should be able to do and achieve these:

- understand how a message can be represented in different ways
- transfer a message using semaphore and Morse code
- appreciate the historic importance of semaphore and Morse code

Need to know

Computers use numbers to represent all the information they hold and process. The precise details of how information is represented is often not that important: we use **abstraction** to hide these details, instead concentrating on the information itself. We can take ideas from how semaphore, Morse, and polybius codes change the representation of information to see the general **pattern** for changing representations as needed.

Key words

Semaphore, Morse Code, transmission, representation

Activities

Prior to the lesson search for or create A4 handout sheets that pupils can refer to, to send messages to a friend across the school hall or classroom.

Time	Teacher activity	Student activity	Resources
20 minutes	Begin the lesson by introducing pupils to the idea of transmitting messages through	Using the handouts, pupils should code and transfer simple messages using semaphore to one another, before their partner decodes	

	semaphore.	them.	
20 minutes	Then progress pupils to looking at and communicating via Morse Code.	Using the handouts, pupils should code and transfer simple messages using Morse code to one another, before their partner decodes them.	
20 minutes	Conclude this lesson by leading an investigation on how semaphore has been used through history	Ask pupils to investigate how Morse code has been used through history. Ask pupils to share any interesting facts they find out.	Information on Napoleon's *le systeme Chappe*

Variations, differentiation, and extensions

From my experience it is a good idea to use torches with pupils rather than sound devices because pupils struggle to make out what their partner is transmitting Morse code.

If pupils are struggling with receiving Morse signals, the teacher may need to synchronise transmission and reception of messages by clapping a slow and steady beat.

Success criteria and assessment

The main success criteria for this activity are the ability to send and receive messages and the understanding that information can be represented in different ways. So long as the pupils understand the idea of changing representations, don't worry too much if they're unable to send long messages or even send any message reliably!

Taking it further...

… into networks

Returning to the hook for teaching encryption, that is, passing messages around the classroom, the pupils can consider encryption in the context of the everyday activities which depend on computer networks.

There are some great free unplugged resources published by the Digital Schoolhouse project called Networks and Communications Unplugged. These activities can be extended by eavesdropping on the messages being passed. From experience, by revisiting these unplugged networking activities after completing encryption activities described above can be a really valuable consolidation learning experience.

… into programming

You can bring computers into these activities. Spreadsheets can be used to encrypt, decrypt, and crack messages. This can be a useful way to teach pupils the Digital Literacy (functional IT skills). Using these tools highlights the value of computers at modeling good thinking and processing information effectively. You can also create encryption and decryption scripts in whatever programming environment the children are using.

A menu of ciphers

The activity resources included in the cipher overviews below are for illustration purposes. They can be either used as standalone activities to develop the computational thinking skills needed before starting of the spy challenges described above. Alternatively, you could decide to combine your study of cryptography with data handling and solving a murder from the book *Certain Death* by Tanya Landman!

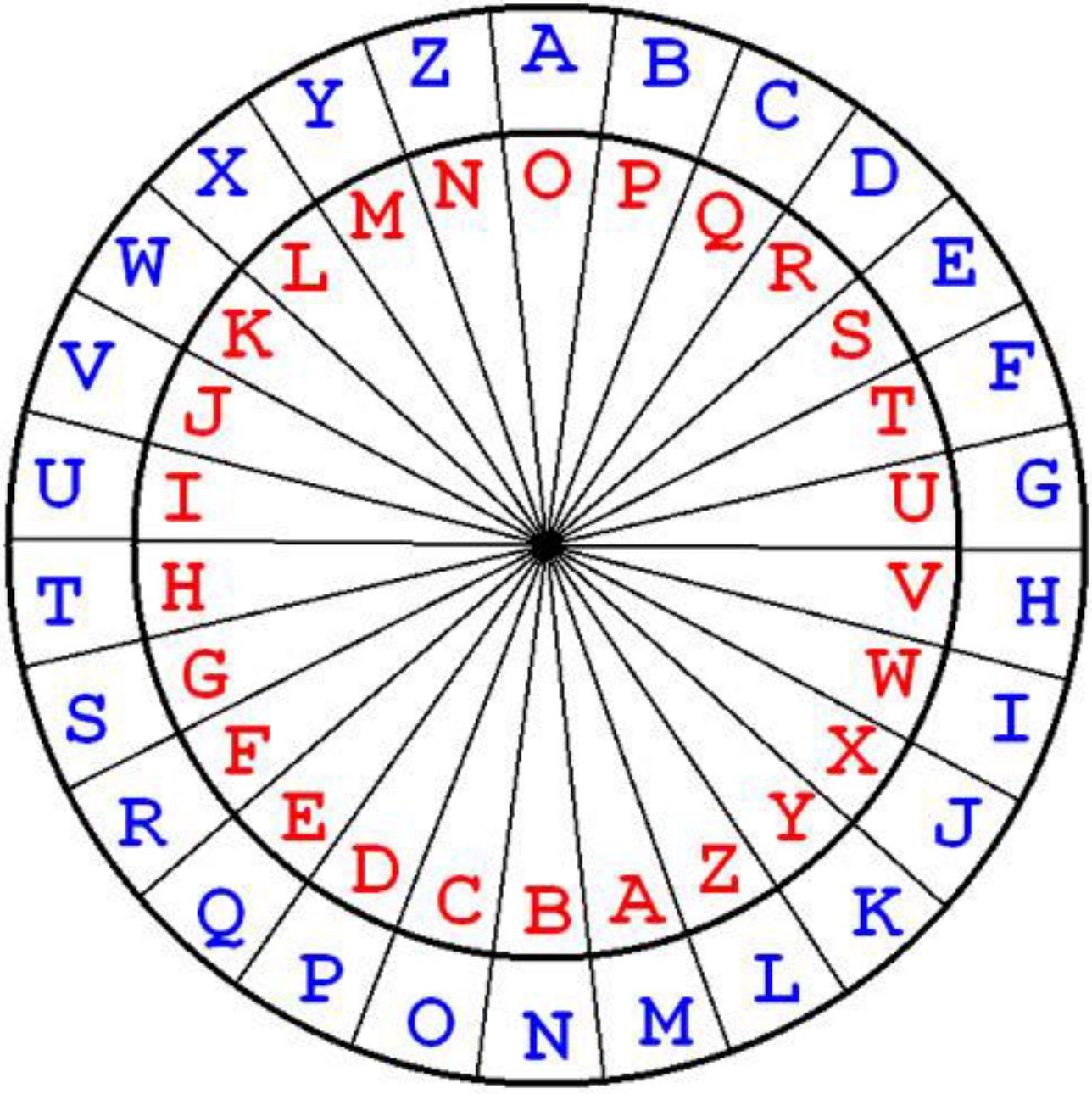
Substitution ciphers encrypt the plaintext by substituting each character of the alphabet for another character. There are a lot of substitution ciphers: I've described the Caesar, pigpen, and polybius ciphers here. **Transposition** ciphers scramble the letters of the plaintext into an anagram. Reverse, railfence and column ciphers are simple examples.

The ciphers given below are generally in their simplest forms. Extension: use a keyword to scramble the ciphertext alphabet. This means the key has two parts: the keyword and the position of the wheel. This works best when the cipher is written out in columns of letters. The keyword is used to scramble the order of the letters on the ciphertext alphabet, so the keyword “SECRET” would mean the ciphertext alphabet starts as “SECRTABDFGHIJKLMNOPQUVWXYZ” before it is shifted.

Caesar Cipher

In a Caesar cipher, each letter in the plaintext is substituted with the character that is one (or more) characters ahead of the character, that is, the character “A” become the character “B” and “E” becomes “F”. For example: “Code Breakers” becomes “ “Dpef Csfblfst”. The key for this cipher is the number of positions to move along when enciphering: a key of 14 would turn "Code Breakers" into "Qcrs Pfsoysfg".

Pupils really enjoy making and using the Alberti Cipher wheel. It is a great ways of supporting pupils with the substitution process when encrypting and decrypting messages. The wheel has two rotating discs, one inside the other, with the alphabet on both, enabling the substitution to take place. To encipher a letter, find the letter on the outer disc and write down the corresponding letter on the inner disc. Deciphering goes from the inner wheel to the outer wheel. For more able pupils, instead of using a cipher wheel, you could ask pupils to create a table with two columns to complete the substitution in the same way as the wheel. In presenting the substitution method in table format, it provides the basis for modeling the cipher in a spreadsheet software.



An Alberti cipher wheel http://www.cs.mtu.edu/~shene/NSF-4/Tutorial/VIG/FIG-Disk.jpg

Making a paper cipher wheel: https://inventwithpython.com/hacking/chapter1.html

Student activity 1 resource (wheel):
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%202%20-%20Code%20wheel.docx

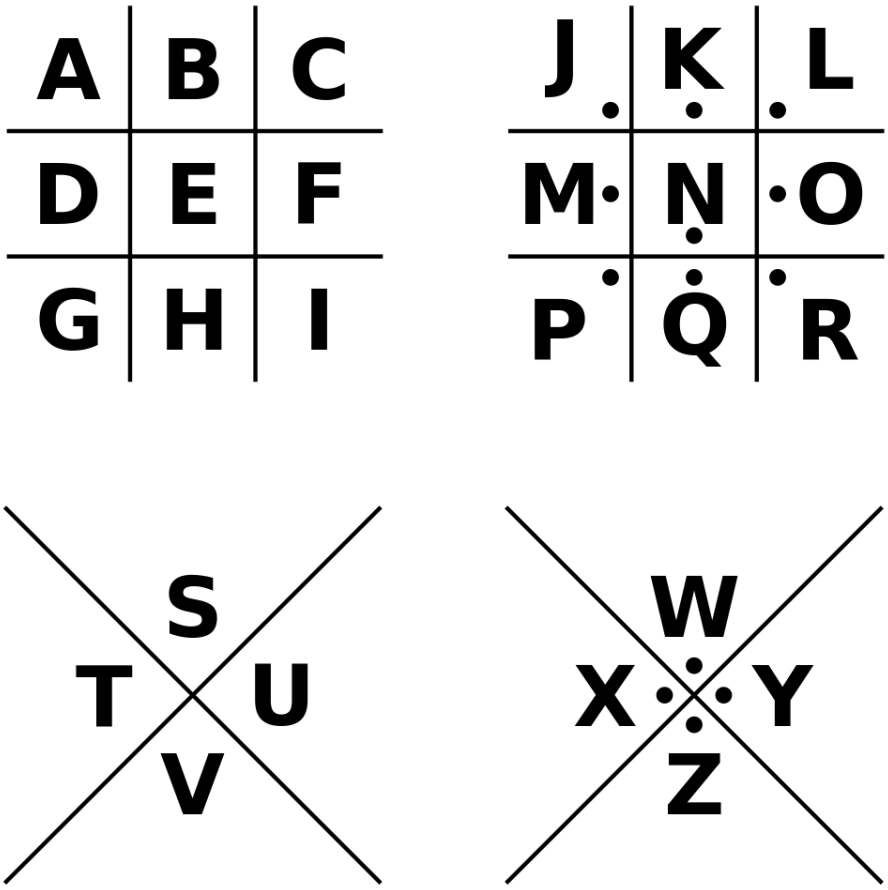
Student resource (activity):
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%202.docx

Resource answers:
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%202%20Answers.docx

Pigpen cipher

The pigpen cipher is another type of substitution cipher. The characters are encrypted by transforming them into symbols. These characters are placed into different grids. Each grid location with different shapes inside, creates a unique set of symbols for the entire alphabet.

The ciphertext is a series of symbols. To decrypt the ciphertext the reader must know the combination of grids and shapes used and how the characters are allocated to the grids.



Pigpen cipher.

Image from
https://upload.wikimedia.org/wikipedia/commons/thumb/3/36/Pigpen_cipher_key.svg/1024px-Pigpen_cipher_key.svg.png

Student resource:
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%205.docx

Resource answers:
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%205%20Answers.docx

Polybius square

A square grid of 5 by 5 is usually used. Although there are 26 characters in the alphabet, two letters are combined into a single cell of the table; usually this is 'I' and 'j'. Alternatively, pupils could experiment with a 6 by 6 grid of 36 characters which could enable them to include numbers (0 – 9) into the grid. The alphabet is allocated to the grid by first populating the first row, and then the second, and so on. Each character in the alphabet is represented by its coordinates in the grid. For example, in a 5 by 5 grid the words "Code Breakers" would become "13 34 14 15 12 42 15 11 25 15 42 43".

To decrypt the ciphertext you need to know the grid used and if any (and which) of the letters of the alphabet have been combined into a single cell. Then use the grid references to turn the ciphertext into plaintext.

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I/J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Polybius square.

Student resource:
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%201.docx

Resource answers:
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/Clue%201%20Answers.docx

Reverse Cipher

The reverse cipher method works by reversing the order of the string of characters in the plaintext to create the ciphertext. For example: "Code Breakers" becomes " "srekaerB edoC". This is possibly the simplest method encryption and the easiest method to crack. You can make it harder to crack by removing both the capitalication of letters and the spaces, for example: "Code Breakers" becomes "srekaerbedoc". Although this is an improvement, it is still relatively easy to crack. To make it a bit harder to crack by further disguising the plaintext, you can do this by grouping the ciphertext string of characters into groups. For example, "Code Breakers" becomes "sre kae rbe doc". To support the pupils with this activity, it is best to give then centimeter squared paper to record the plaintext and ciphertext.

Student activity resource: http://northofsepo.wikidot.com/activity:reverse-cipher

Railfence cipher

When using this method, the message is split between two or more rows. To create the ciphertext you add the characters from the second row to the end of the first row. For example, splitting a message over two rows would be: “Code Breakers” becomes “cdbekroeraes”. Splitting a message over three rows the ciphertext looks quite different with “Code Breakers” becoming “Ceee Obar drks”. To decrypt the ciphertext, to view the plaintext, you need to know the number of rows that the message was split over and then split the continuous string of characters into separate words.

Column cipher

This method involves creating a grid. Enter the message starting from the cell in the first row and column, entering the characters from left to right, before moving onto the second row. The simplest way to encrypt the message and create the ciphertext is to again start at the top left hand corner but this time move down the column before proceeding to the next column to create the anagram. To decrypt the ciphertext, you need to know the grid dimensions and the pattern used to encyrpt the plaintext.

For instance, the phrase CODE BREAKERS, enciphered in a four-column grid, would become CBKOREDEREAS

C	o	d	e
b	r	e	a
k	e	r	s

A column cipher

Student encyption activity resource: https://www.youtube.com/watch?v=2_D2CkteKZI

Student decryption activity resource: https://www.youtube.com/watch?v=yNys2q9xmno

Famous ciphers through the ages

The earliest ciphers found by archaeologists are hieroglyphics carved into Ancient Egyptian monuments (1900BC), then the Ancient Greeks were next to use ciphers to send military messages using the Polybius Square method (800BC), before Julius Caesar used a type of substitution cipher (50BC) which he tattooed into his messengers heads before sending them to his military leaders.

In 1467 the Alberti Cipher was invented in Italy. It was a wheel with two rotating discs, one inside the other, to perform the substitution and encrypt letters in the alphabet. Then in 1586, Mary Queen of Scots had her head chopped off because her communications with her followers were cracked, that is her codes were broken, and it revealed a plot to assassinate Queen Elizabeth I was discovered.

We have all heard about the Enigma code. It was originally intended to be used by finance and banking as a means of transferring money. However, it was further developed and used by the German military to send classified information about military strategies during World War Two. However, whilst the German military were trailing the Enigma code; in 1932, a collaboration between the French Military Intelligence and Polish Cipher Bureau actually cracked the code.

Enigma design meant that there were 159 million million million possible settings to choose from and the cipher key was changed daily. During World War Two, Alan Turing and his

colleagues further developed this work and had some limited success. However, the team at Bletchley Park had to rely upon luck and incredible bravery because it wasn't until a German Enigma operator made a mistake and the daring capture of a book of cipher keys and an Enigma machine were the Bletchley Park team able to regularly fully decrypt the messages. Because there teams at Bletchley Park were working around the clock and against the clock crack the codes, they required a vast amount of computations to be carried out very quickly and humans were just too slow so the Bletchley Park team built the first computer to speed things up. It was called colossus and was designed for the sole purpose of decrypting codes.

Modern day encryption and code breaking using machines was developed at Bletchley Park during World War Two. Therefore, in this chapter we have looked at how we can teach cryptography using a combination of unplugged (without computers) and plugged-in (with a computer) activities.

A useful slide deck for teaching the history of cryptography which is described in the Overview section of this chapter can be found at:
http://www.digitalschoolhouse.org.uk/sites/default/files/cms/docs/6.%20Cryptography.pdf

Reflective questions

1. What is the role of the teacher in this chapter, through developing pupils computational thinking capabilities and the pedagogical approaches described? How is it different to the traditional teaching pedagogy of ICT?

2. How could the pedagogical approaches described in this chapter help to build confidence, resilient, communication and collaboration skills in pupils?

3. Reflecting on the success criteria listed for each Activity, did you see your pupils show the computational thinking techniques lists? Did your pupils demonstrate any others from the list published by Computing At School in the Computational thinking guidance for teachers document?

4. How important is effective Assessment for Learning (AfL) in supporting pupils in developing their computational thinking capabilities? What does a pupil demonstrating 'computational thinking' skills look like? What behaviours will they be displaying during the lessons to help you to assess their learning?

5. What are the challenges for classroom practitioners of pupils developing and applying creativity and computational thinking to help pupils take ownership of both the process they've gone through and the artefacts they produce.

Discussion

The encryption algorithms used here have been too weak for real uses for centuries, but the process of using them and understanding them allows pupils to explore many aspects of computational thinking, and how computers are a tool, rather than the focus of the learning. Once pupils understand a problem and how to solve it, they can move from the unplugged to the plugged, using computers and writing programs to run algorithms faster and on more data than is convenient for humans.

The Computer Science Teachers Association (CSTA) suggest that five dispositions are developed through computational thinking. From my classroom experience, I

would agree that the following attitudes can be observed in learners during Computing lessons:

- Confidence in dealing with complexity
- Persistence in working with difficult problems
- Tolerance for ambiguity
- The ability to deal with open-ended problems
- The ability to communicate and work with others to achieve a common goal or solution

Like any thinking skill, it is important to provide learners with opportunities to develop them. In the activities, I've suggested how to differentiate the tasks to support a range of pupil abilities. Gradually, as pupils progress through the activities in this chapter, it provides for more pupil-lead learning but the key questions employed by the class teacher are key to prompting and steering the pupils. When these computational thinking skills have been mastered, pupils should be able to successfully apply them to a range of problem solutions – not just encryption. Therefore, it is reasonable to suggest that there is an association between the pupil, the activity, and the way in which the activity are presented to pupils by the class teacher.

Summary and key points

Although this chapter primarily deals with data representation and a range of simple algorithms to encrypt and decrypt text, it also highlights how this topic can be made fun and engaging for all pupils by adapting our pedagogical approaches to embed computational thinking through a constructionist approach to teaching. But also highlights how relating the topic to the world that they, the pupils, understand, will serves as a hook to help them to understand the 'real world' and technologies that they take for granted. Pupils discover that all new technologies are iterations of previous technologies through investigating semaphore and Morse Code as a representation for transferring data and by looking at there historical relevance.

Finally, the pedagogical approach described through the activities presented in this chapter rely upon classroom practitioners developing both their subject content knowledge, but just as importantly, their pedagogical content knowledge which comes with experience and being a reflective practitioner.

Resources

Non commercial:

CS Unplugged (2015). Public Key Encryption. Available here: http://csunplugged.org/public-key-encryption/ [Last accessed: 01/01/2016]

CS Unplugged (2015). Cryptographic Protocols. Available here: http://csunplugged.org/cryptographic-protocols/ [Last accessed: 01/01/2016]

CS Unplugged (2015). Scout Patrol (Encryption). Available here: http://csunplugged.org/scout-patrol-encryption/ [Last accessed: 01/01/2016]

Digital Schoolhouse. (2014). Networks and communications unplugged. Available here: http://community.computingatschool.org.uk/resources/2528 [Last accessed: 01/01/2016]

Digital Schoolhouse (2013). Cryptography: Secrets, Secrets, Secrets. Everyone has them! Available here: http://www.digitalschoolhouse.org.uk/workshops/cryptography-secrets-secrets-secrets-everyone-has-them [Last accessed: 01/01/2016]

Digital Schoolhouse (2013). Step by step tutorials for modeling ciphers in a spreadsheet software. Available here: https://www.youtube.com/results?search_query=mark+dorling+encryption [Last accessed: 01/01/2016]

Crypto Corner. (2016). Downloadable Resources. Available here: http://crypto.interactive-maths.com/downloadable-resources.html [Last accessed: 01/01/2016]

Simkin. M., (2006). Using spreadsheets to Teach Data Encryption Techniques. Available here: http://www.aisej.com/doi/pdf/10.3194/aise.2006.1.1.27 [Last accessed: 01/01/2016]

Commercial:

Berry, M., (2015). Switched On Computing - Year 5 Unit 2: We are cryptographers. London: Rising Stars.

Dorling, M., and Rouse, G ed. (2014). Compute-IT Series (Teacher and Student books): Book 3 Unit 1: Cracking the code. London: Hodder Education.

Further reading:

BBC. (2013). BBC Code breakers: Lost heros. Available here: http://www.bbc.co.uk/programmes/b016ltm0 [Last accessed: 01/01/2016]

BBC. (2013). BBC Megabits: How computers changed the Second World War and all future digital communications. Available here: http://www.bbc.co.uk/programmes/p011lptc [Last accessed: 01/01/2016]

Bletchley Park. (2016). Learning At Bletchley Park. Available here: http://www.bletchleypark.org.uk/edu/ [Last accessed: 01/01/2016]

Brennan, K. and Resnick, M., (2012) New frameworks for studying and assessing the development of Computational Thinking'. Available here: http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf [Last accessed: 01/01/2016]

Computing At School, (2016). Computational thinking: A guide for teachers. Available here: http://community.computingatschool.org.uk/resources/2324 [Last accessed: 01/01/2016]

Computing At School, (2016). QuickStart Computing, Section 4 Teaching. Available here: http://www.quickstartcomputing.org/secondary/section4.html [Last accessed: 01/01/2016]

CSTA (2015), What does Computational Thinking develop in learners

http://www.csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf

Department for Education. (2013). National Curriculum in England: Computing programme of study. Available here: https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study [Last accessed: 01/01/2016]

Department for Education (2015) Commission on Assessment Without Levels. Available here: https://www.gov.uk/government/publications/commission-on-assessment-without-levels-final-report [Last accessed: 01/01/2016]

Department for Education (2015) Government response to the Commission on Assessment Without Levels. Available here: https://www.gov.uk/government/publications/commission-on-assessment-without-levels-government-response [Last accessed: 01/01/2016]

Dorling, M. (2015). Networks and communications. In: Williams, M. ed. Introducing Computing: A guide for teachers. Oxon: Routledge. p107-120

Dorling, M., and Woollard, J. (2015). Planning and assessing computing and computational thinking. In: Allsop, Y., and Sedman, B,. ed. Primary Computing in Action 2015. John Catt Education Ltd. P: 163- 184.

Landman, Tanya. (2013) Murder Mysteries 6: Certain Death, Walker Books. http://www.walker.co.uk/Murder-Mysteries-6-Certain-Death-9781406347432.aspx (though you might have to go hunting for second-hand copies)

Ofsted (2015) School inspection handbook from September 2015. Available here: https://www.gov.uk/government/publications/school-inspection-handbook-from-september-2015 [Last accessed: 01/01/2016]

Papert, S. and Harel, I., (1991) Situating Constructionism'. Available hère : http://www.papert.org/articles/SituatingConstructionism.html [Last accessed: 01/01/2016]

Royal Society (2012) Shutdown or Restart? Available at: https://royalsociety.org/topics-policy/projects/computing-in-schools/report/ [Last accessed: 01/01/2016]

Scratch Ed, How do I assess the development of Computational Thinking. Available here: http://scratched.gse.harvard.edu/ct/assessing.html [Last accessed: 01/01/2016]

Scratch Ed, Assessing development of computational practices. Available here: http://scratched.gse.harvard.edu/ct/files/Student_Assessment_Rubric.pdf [Last accessed: 01/01/2016]

Selby, C. & Woollard, J. Computational Thinking: The Developing Definition. http://eprints.soton.ac.uk/356481/ [Last accessed: 01/01/2016]

Selby, C., Dorling. M, & Woollard, J. Evidence of Assessing Computational Thinking http://eprints.soton.ac.uk/372409/1/372409EvidAssessCT.pdf [Last accessed: 01/01/2016]

Wing, J. (2006) Computational Thinking, Communications of the ACM, Available here:

https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf [Last accessed: 01/01/2016]

Wing, J (2011). Research Notebook: Computational Thinking - What and Why?
http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why
[Last accessed: 01/01/2016]