Efficiency of algorithms

There is often more than one way to write an algorithm. Sometimes one way is more efficient

Efficiency relates to use of computing resources and **time taken** to complete the algorithm. A **quicker** algorithm is **more efficient**

This often comes down to **number of lines of code**

Efficiency of algorithms

3.1.2 Efficiency of algorithms

Content	Additional information	Chk
Understand that more than one algorithm can be used to solve the same problem.		
Compare the **efficiency** of algorithms explaining how some algorithms are more efficient than others in solving the same problem.	Formal comparisons of algorithmic efficiency are not required. Exam questions in this area will only refer to time efficiency.	

Time efficiency = how many lines of code have to execute

Efficiency of algorithms

When loops are involved, the number of lines of code needed to run the algorithm isn't always clear

In general, **indefinite iteration** (while loop) will usually be **more efficient** than **definite iteration** (for loop)

Efficiency of algorithms

e.g. look for "o" in "wolloomooloo"

- While loop needs to look at two characters before it finds the o. Iterates **twice**
- For loop **has** to look at every letter - even though it found it quickly. So it needs to iterate the loop **12 times**

The logic for a while loop is often a bit more complex and more lines of code might be needed.

Efficiency of algorithms

But look for "k" in "wolloomooloo"

- While loop needs to look at 12 characters. Iterates **12 times** - because it never finds "k"
- For loop **has** to look at every letter, so it needs to iterate the loop **12 times**

In this case there's no difference in iterations, but the logic for the while loop is probably a little more complex…

Efficiency of algorithms

- many problems have more than one algorithm as a solution

- some algorithms are more efficient than others

- efficiency is measured by how long it takes to run the algorithm (lines of code)

- while loops will end sooner than for loops in many case. This generally makes them more efficient