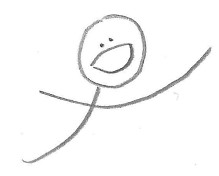
Read the following and answer the questions in the text.

You are a mathematical inventor, and you have just come up with a wonderful new mathematical machine! This machine is able to rearrange sequences of 𝑛 letters, and you have decided to call it an 𝑛-*permutator*. An example of a 7-permutator is below:

1 2 3 4 5 6 7

ቀ ቁ

3 6 2 4 7 1 5

It works as follows. Suppose you have a sequence of seven letters:

A B C D E F G

For each number 𝑖 in the top row of the 7-permutator, the letter in position 𝑖 gets moved to the position indicated by the number underneath 𝑖. So the letter A gets moved from position 1 to position 3, B from position 2 to position 6, etc. After the permutator above is applied to all of the letters, the new arrangement looks like this:

F C A D G B E

Soon, word gets around about your magnificent invention – and people start asking you for help!

Diagram, engineering drawing

Description automatically generatedText, letter

Description automatically generatedDiagram, engineering drawing

Description automatically generated

As more and more people come to you with scrambled words that need restoring, you begin to realise that your invention may have some business potential! You decide to pitch your idea to some investors. But you feel that before doing so, it would be prudent to do some investigation and research about 𝑛-permutators and their properties. (Luckily, having recently studied counting techniques in the Year 11 Specialist course, you are well-prepared for the task.)

To start with, you focus on the case with 𝑛 = 7. An obvious first question is:

1. What is the total number of different 7-permutators? **7! = 5040**

You’ve noticed that very often, people come to you with a word that has just two letters swapped, and so require an 𝑛-permutator that swaps those two letters back again. So next you try to answer the following question:

1. How many different 7-permutators swap the letters within just one pair (and leave all the other letters in the same place)? - **7C2 = 21**

Some further questions which occur to you are the following:

1. How many different 7-permutators swap letters within exactly 2 pairs? **(7C2 \* 5C2)/2! = 105**

How many swap letters within 3 pairs (always leaving the other letters in the same place)?

**(7C2 \* 5C2 \* 3C2)/3! = 105**

1. How many different 7-permutators move *exactly* 3 letters (leaving the other letters where they are)? **7C3 \* 2! = 70**

Text, letter

Description automatically generatedHaving crunched these numbers, you present your idea to some investors. One of them is very interested, but has a question:

“It seems that some 𝑛-permutators offer better value for money than others. For instance, the following 7- permutator returns all letters to their starting positions after just two applications:

1 2 3 4 5 6 7

ቀ ቁ

5 3 2 7 1 6 4

But the one below can be applied *seven* times before the order is the same as it was originally:

1 2 3 4 5 6 7

ቀ ቁ

2 3 4 5 6 7 1



This is because it rotates the seven letters around in a cycle – it’s like seven permutators in one! I’d like to know more about these ‘high value’ permutators before I stump up the cash!”

**Definition**: the *loop length* of a given 𝑛-permutator is the minimum number of times (greater than 0) that it must be applied to a sequence of 𝑛 distinct letters in order to return them all to their original positions.

The potential investor would like answers to the following questions:

1. Can you find examples of 7-permutators which have loop lengths of 3, 4, 5 and 6?
2. Are there ‘different ways’ of obtaining 7-permutators with a given loop length (such as 6)?

**You can have one loop of 6 (1, 2, 3, 4, 5, 6 🡪 2, 3, 4, 5 ,6, 1, 7)**

**or a loop of 2 and 3 (1, 2, 3, 4, 5, 6, 7 🡪 2, 1, 4, 5, 3, 6, 7)**

1. What is the total number of 7-permutators with a loop length of 7? - **6! = 720 (Arranging a 7 loop)**
2. What is the total number of 7-permutators with a loop length of 5? **7C5 (number of loops of 5) \* 4! (Arrangements within those loops = 504)**
3. What is the maximum possible loop length of a 7-permutator? **4 loop \* 3 loop = 12 loop**

Having figured out answers to these questions, you tell the potential investor, who says:

“I’m very impressed. But there’s one more thing I’d like to know. So far you’ve told me all about 𝑛- permutators with 𝑛 = 7. I presume most people won’t have need of an 𝑛-permutator with 𝑛 more than about 15. I’d like to know if the methods for answering these questions can be generalised for values of 𝑛 up to 15. Can you investigate please?”

Bajay Bisnath’s Epic Notes

Shape

Description automatically generated

* Example of a 7-loop cycle
  + This can exist in any permutator equal to or greater than 7

How many different 7-permutators move exactly 4 characters (leaving the other characters where they are)?

* + First choose the characters being moved (7C4), now you can either have a 4 loop, or two 2 loops. A 4 loop has 3! combinations for every set of 4 characters. For the two 2 loops. First choose the characters in the first loop (4C2) which incidentally decides the second loop with the remaining characters. Because both the first and second loop can be identical

(e.g. [1, 2][3, 4] is the same as [3, 4][1, 2]) you must divide by two. Then multiply these values by your 7C4 to get the final result!

For max loops:

* 6 is the first loop length that can be made in multiple ways:
  + You can have a loop of 6 (Numbers move along 6 places to get to their starting position)
  + Or a loop of 2 and 3 (Numbers need to move around 2\*3 times to get to their starting position)
  + 6 is the smallest lowest common multiple of 2 co-primes greater than 1!

For an n-permutator of 15, what is the maximum loop length?

* Is it 7 \* 3?
  + 7 + 3 > 15, 7 \* 3 = 21 therefore is valid BUT!!
* 8 \* 7 is larger!
  + 8 + 7 = 15, 8 \* 7 = 56, but wait there’s more!!
* 3 \* 4 \* 5!
  + 3 + 4 + 5 = 12, 3 \* 4 \* 5 = 60, this is the maximum loop length for an n-permutator of 15, but this is also the maximum loop length for an n-permutator of 12 :D
* Your values must add to be equal to or less than your n-permutator value (15 in this case, otherwise known as ‘n’). They must all be co-primes meaning they do not have a higher common factor than 1.
  + If you use 4, then you can’t use 2 because it will be meaningless, same for using 9 and 3.
    - 4 \* 5 \* 9 loop length is the same as 3 \* 4 \* 5 \* 9 loop length as the 3 and 9 are not coprime and the 3 is cancelled out!

What is the total amount of 7-permutators with loop length of 6?

* You must add both the loop of 6 and the loop of both 2 and 3 that become a loop length of 6 together!
* 7C6 \* 5! + 7C2 \* 5C3 \* 2! = 1260
  + There is 5! way of ordering a loop of 6 (7C6)
  + 7C2 denotes loop of 2, 5C3 denotes loop of 3 using the remaining characters, 2! Is the number of ways to arrange the loop of 3 [1, 2, 3], [1, 3, 2]
    - (Remember to think of these as a cycle like above!)
    - This works if you do 7C3 \* 5C2 \* 2! as well