# MA3K7 Week 2 Rubric

Rizwan (2107025)

## 1 Entry

I KNOW

What i know so far is that this problem involves a sequence and i need to find out how to carry on the chain until it starts looping and think how many of these different sequences I can produce.

SPECIALISE

So from this I want to be able to see if I can find a pattern. and be able to use this pattern to be able to artificially produce sequences and soon be able to use mathematical knowledge of combinatorics to be able to yield all the different chains that I am able to get.

$$1 \to 5 \to 6 \to 1 \to 7 \to 8 \to 5 \to$$

We can simplify this chain into just a normal sequence. Then we can work from here. A couple of questions come to my mind when I think of this chain. For example, the rules that govern this sequence. looking at it first hand, it doesn't seem to have a direct rule at all but during the attack phase I will look at it further. Another potential question relating to the first one is the potential fro this sequence to loop infinitely. If this sequence was to loop infinitely then that would make it a lot harder to dissect the problem as further questions will pop up. For example:

- Is it dependent on the starting numbers?
- Do finite loops exist for this particular problem?
- If so, then what are the conditions for finite loop?

and so on. All of these questions will pose further problems and further analysis will be required to answer all of these questions. Another potential question is if this has finitely amount of different chains. Due to the constricted nature of the problem this thought wasn't really entertained as it would be impossible.

Another thing that interested me about this problem was how to define a stopping point. With the nature of the problem and the my initial thought of infinite chains, it begs a question of how and what are the conditions for a stopping point in the chain. For example:

- Does their exist certain numbers that need to appear consecutively to yield a loop?
- Are these numbers dependent on the starting numbers or are they different depending on the chain?

## 2 Attack

Looking at the problem initially had me clueless as it seemed like their was no pattern present at all. Using my knowledge of arithmetic and geometric sequences

STUCK

I could not find any pattern that worked.

$$1 \rightarrow 5 \rightarrow 6$$

I then went deep into the sequence to see if their were any absurd patterns I didn't Aha! consider before. Then I noticed the first 3 terms.

1+5 = 6 so then I thought could this be a modification of the Fibonacci sequence so I then carried on the sequence and saw it wasn't as easy as I initially anticipated.

$$5 \rightarrow 6 \rightarrow 1$$

5+6 = 11 and what is written here is 1. Then another thought sprang into my Aha! mind.the last digit is the same as 1 and then I went to check the rest of the sequence and the trend continued that the last digit is the same as the sum. So then I thought. MOD 10. Its a Fibonacci sequence with mod 10 applied to it.

I then thought to carry on this first first sequence to see if I can finish off this sequence. Below is the final result of this initial sequence(I used a bit of Python code to generate the sequence which I have linked at the end).

[1, 5, 6, 1, 7, 8, 5, 3, 8, 1, 9, 0, 9, 9, 8, 7, 5, 2, 7, 9, 6, 5, 1, 6, 7, 3, 0, 3, 3, 6, 9, 5, 4, 9, 3, 2, 5, 7, 2, 9, 1, 0, 1, 1, 2, 3, 5, 8, 3, 1, 4, 5, 9, 4, 3, 7, 0, 7, 7, 4]

In order for the sequence to repeat itself 1 and 5 need to consecutively be next to each other. This is only possible when n-2 term is 7 and the n-1 term is 4. Which would then cause 1 and 5 to repeat again and then the whole sequence will repeat.

Playing around with these sequences we can see a pattern. The above example yielded that after 60 terms the sequence repeats. If we take 6,8 as the start we get,

[6,8,4,2,6,8]

which repeats after 6(a factor of 60). another example is if we start with 2,2 as the start,

[2,2,4,6,0,6,6,2,8,0,8,8,6,4,0,4,4,8,2,0,2,2] which repeats after 20 terms(another factor of 60). After messing about with a few more examples we can conclude that these chains will repeat themselves after a certain factor of 60(which are 1,2,3,4,5,6,10,12,15,20,30,60).

Another example is if we use 5,4 as the starting numbers we get, [5, 4, 9, 3, 2, 5, 7, 2, 9, 1, 0, 1, 1, 2, 3, 5, 8, 3, 1, 4, 5, 9, 4, 3, 7, 0, 7, 7, 4, 1, 5, 6, 1, 7, 8, 5, 3, 8, 1, 9, 0, 9, 9, 8, 7, 5, 2, 7, 9, 6, 5, 1, 6, 7, 3, 0, 3, 3, 6, 9] which in 60 terms repeats itself which further emphasises my point. I also detected another trend while experimenting with different sequences.

If you start with two even numbers then the rest of the sequence will be even and the exact same principle would occur if we had started with 2 odd numbers. This led me to think. In these specific cases, would the sequence converge faster due to the lack of options(as there are only 4 even numbers and 5 odd numbers. I didn't delve deep in this thought as I felt it didn't really help me much.

Now to find out how many different chains we have, we do the following:

First we have to check if the order does matter for sequences. Lets take the example that was given to us and switch it around:

5,1,6,7,3....

as we see the terms do change slightly after the first few terms so therefore order does in fact matter for this case. Due to this we don't need to remove any of the sequences that have the same number but switched.

My initial guess was that we take all sequences that start with 0 and any other number between 0 to 10(but not including). So we take all sequences of the following form which start of as (0,0),(0,1),(0,2)....(0,9).

Since (0,0) regardless of whether you switch them will give you the same sequence. So with the sequences starting with we have 10. This also applies to every other number between 1 and 10. so we can include the sequences (1,0),(1,1),(1,2) etc.

Using Python, I then created a grid where it modelled the starting 2 numbers of all possible chains:

```
(0, 0) | (0, 1) | (0, 2) | (0, 3) | (0, 4) | (0, 5) | (0, 6) | (0, 7) | (0, 8) | (0, 9)
-------------------------------------------
(1, 0) | (1, 1) | (1, 2) | (1, 3) | (1, 4) | (1, 5) | (1, 6) | (1, 7) | (1, 8) | (1, 9)
-------------------------------------------
(2, 0) | (2, 1) | (2, 2) | (2, 3) | (2, 4) | (2, 5) | (2, 6) | (2, 7) | (2, 8) | (2, 9)
-------------------------------------------
(3, 0) | (3, 1) | (3, 2) | (3, 3) | (3, 4) | (3, 5) | (3, 6) | (3, 7) | (3, 8) | (3, 9)
-------------------------------------------
(4, 0) | (4, 1) | (4, 2) | (4, 3) | (4, 4) | (4, 5) | (4, 6) | (4, 7) | (4, 8) | (4, 9)
-------------------------------------------
(5, 0) | (5, 1) | (5, 2) | (5, 3) | (5, 4) | (5, 5) | (5, 6) | (5, 7) | (5, 8) | (5, 9)
-------------------------------------------
(6, 0) | (6, 1) | (6, 2) | (6, 3) | (6, 4) | (6, 5) | (6, 6) | (6, 7) | (6, 8) | (6, 9)
-------------------------------------------
(7, 0) | (7, 1) | (7, 2) | (7, 3) | (7, 4) | (7, 5) | (7, 6) | (7, 7) | (7, 8) | (7, 9)
-------------------------------------------
(8, 0) | (8, 1) | (8, 2) | (8, 3) | (8, 4) | (8, 5) | (8, 6) | (8, 7) | (8, 8) | (8, 9)
-------------------------------------------
(9, 0) | (9, 1) | (9, 2) | (9, 3) | (9, 4) | (9, 5) | (9, 6) | (9, 7) | (9, 8) | (9, 9)
-------------------------------------------
```

This yielded that there are 100 different chains that can be produced with these conditions provided in the question.

However, since we form a bracelet there comes a possibility that some of these sequences, regardless of the starting numbers, are the same because if you 'loop' around the sequence formed could be the same as another sequence but shifted so we need to account for this.

I created a python script that accounted for all the possible starting numbers that are the same as the one bracelet formed. For example, the (6,2) bracelet is the exact same as the (0,4). My thought process was to remove the sequences with the starting terms that are consecutive in the chain. For example, in the chain (6,2) the first few terms are 6,2,8,0,8,.... etc. Using these terms we can conclude that the (2,8) will also form the same bracelet as (6,2). I then looped through until i covered all the sequences and then counted the amount of iteration.

I concluded that there are 6 different bracelets in all 100 starting points.

# 3 Review

During this problem, it required at lot of out of the box thinking and often weird thought processes to try and get a solution. Using Python really helped me deduce key information about how to attack this problem.

For example, finding out when the sequence starts to repeat itself. Python really helped me simplify a lot of mathematics which otherwise would've been quite a pain to calculate. For example the sequence lengths were extremely unpredictable at first but then once I found a common pattern it made it a lot easier and less daunting.

REFLECT

Despite that this was a really fun problem for me to get started on and I look forward to more!!

## Supplementary material

Below is attached my GitHub repository where my code resides:

`https://github.com/rizwan3254/PS-with-yth`