Software Developer Problem

Problem definition

Given a list of integers, identify sequences where successive numbers exactly *N* indexes apart have a value equal to *N* multiplied by the previous number in the sequence.

Rules:

- *N* must be greater than 1
- Sequences with less than 3 entries should be ignored
- Sequences returned must always be the longest possible for a given value of *N*
    - In other words, there is max one resulting sequence for any value of *N*
- Sequences of *all* zeros do not count

Example

Consider the following list of integers:

Number	2	10	4	3	8	6	9	9	18	27
Index	0	1	2	3	4	5	6	7	8	9

The following sequences are found:

- 2, 4, 8 (for *N=2*)
- 3, 9, 27 (for *N=3*)

Exercise

Write a Kotlin/Java program that accepts a list of Integers as an argument, and returns a list of sequences matching the required pattern, with each sequence being a list. The method signature must be as follows:

```
fun findSequences(input: List<Int>): List<List<Int>>
```

OR

```
List<List<Integer>> findSequences(List<Integer> input)
```

Then run the program with this input list:

> 2,10,4,3,8,6,9,9,18,27,1,52,81,10,1,0,2,0,4,0,8,0,16,0,32,0,64,0,128,2,10,4,3,8,6,9,9,18,27,1,52,81,10,100,50,0,0,0,0,0,0

Providing your solution

Your answer should be returned as an email which contains your source code, and any other files, as .kt or .java files in a ZIP file.

Additionally in the body of the email please:

- Provide the list of sequences that you found for the input, specifying the value of *N* each sequence was found for
- State the sum of all integers in each sequence
- State the combined sum of all sequences
- State the algorithmic complexity of your solution using big O notation