You are given a list of intervals of the form:

(shipment, vessel, sourcePort, destinationPort, start, end)

We would like to call an external API which has the following signature:

getHistoricalAisData(vessel, start, end)

This API returns telemetries (data points) for every hour in the given time range. We pay for every returned data point, and for excess calls.

Write an algorithm to produce a set of API calls that will cover each data point once, while making as few calls as possible.

Example:

(A, 1, NY, ANWERP, 1/1, 14/1)

(A, 2, ANTWERP, HAIFA, 14/1, 20/1)

(B, 2, ANTWERP, HAIFA, 14/1, 20/1)

(B, 3, HAIFA, ASHDOD, 20/1, 21/1)

(C, 2, HAIFA, ANTWERP, 20/1, 26/1)

(D, 3, ASHDOD, KOBE, 24/1, 10/2)

Output:

(1, 1/1, 14/1)

(2, 14/1, 26/1)

(3, 20/1, 21/1)

(3, 24/1, 10/2)

Design the program. Break it down to functions/methods as needed. Can you reduce the problem to a smaller well-defined algorithmic problem?

Design the Algorithm.

What is the time and space complexity of the algorithm?

Implement in Java or Kotlin.

Enjoy!