QUANTUM BREAKDOWN

Catalysts Coding Contest
3rd April 2020

data by Spire   event organizer   cloudflight

December 2028, Earth

The world is now a fundamentally different place.
Advancements made in the field of **quantum computing** made decryption a trivial problem. Without any proper encryption techniques yet discovered, the world is forced to resort to other, at least for now, more old-fashioned ways of **transfering critical data** from **point A to point B.**

You're hired by the middle-eastern security magnate *So-Loh Mid* to develop such a system of data passing as to ensure complete security and data integrity.

data by Spire   event organizer   cloudflight

> **Your task**

› Find an efficient and fail-safe way to transfer data between points

› The internet is not an option anymore as encryptions are not safe

› You'll have to find another more... physical way to transfer data!

› You'll be assisted by So-Loh Mid's Russian CIO, Kerry Kotl

data by Spire   event organizer   cloudflight

LEVEL 1

> Level 1

Kerry Kotl:
Let's make one thing clear. I don't give you mana until you fix issue. Here, look over these flight charts, see what you can learn from them. I won't tell you my plan until you can at least prove to me you're up to the task.

Task for Level 1: Find the boundaries of the map, maximum altitude and time span

data by Spire   event organizer   cloudflight

> Level 1

	Input	Output
Format	*N* *timestamp,lat,long,altitude* *(repeats N times)*	*minTimestamp maxTimestamp* *minLat maxLat* *minLong maxLong* *maxAltitude* Self explanatory. Find the minimum and the maximum for each type of value, except for the altitude where we already know the minimum is 0 (generally, but the slice of data you see might not have it). Float values are compared with a **maximum allowed error of 10⁻⁵**
Types	**N (int)** Number of flight entries that follow **timestamp (int)** Amount of seconds since a fixed point in time **lat (float)** North latitude of the coordinate. In degrees **long (float)** East longitude of the coordinate. In degrees **altitude (float)** Meters above the sea level	**minTimestamp (int)** **maxTimestamp (int)** **minLat (float)** **maxLat (float)** **minLong (float)** **maxLong (float)** **maxAltitude (float)**
Example	*5* *136932,48.297,16.503,1379* *144068,45.262,39.702,1866* *212782,41.287,2.089,0* *370277,43.959,21.427,11582* *578963,51.531,19.923,10058*	*136932 578963* *41.287 51.531* *2.089 39.702* *11582*

Float values are compared with a **maximum allowed error of 10^{-5}**

Good luck!