
LEVEL 4

> Level 4

Kerry Kotl:

Yes, yes, but now let's focus on those planes again. Can you pinpoint their location given a certain timestamp?

You are given an aircraft coordinates in time

Task for Level 4: Find the lat, long, and altitude of a plane at a given timestamp

> Level 4

- Downloading the whole flight map for each test case would be too much. We've preprocessed some data and separated each individual flight in its own file. In the level archive, you can find another archive called usedFlights. There you will find multiple ".csv" files with integers as filenames.
- The level archive can be found in CatCoder under the name of "Input file", next to the level description.
- In the testcase, we will refer to those flight files by their corresponding integer (a.k.a flight Id). That means that you have to open the file with that name, load its flight course into memory and then do additional work on it.
- This will allow us to use a flight for more than 1 test case and not have you download it multiple times.
- **Please do not delete the files you download for each level**. The next levels will require them as well.

> Level 4

	Flight file
Format	*startAirport* *endAirport* *takeOffTimestamp* *N* *timestampOffset,lat,long,altitude (repeats N times)* *Entries are sorted by their timestampOffset.*
Types	**originAirport (string)** IATA code of the origin airport **destinationAirport (string)** IATA code of the destination airport **takeOffTimestamp (int)** Timestamp of the moment the plane took off **N (int)** Number of coordinates that follow **timestampOffset (int)** Number of seconds since takeoff **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
Example	*EBBR* *EDDT* *386905* *280* *0,50.903,4.481,0* *589,50.878,4.429,1280* *599,50.873,4.427,1341* *611,50.861,4.422,1455* *620,50.857,4.42,1493* *629,50.852,4.42,1516*

> Level 4

- As you can see in those files, the flight course is not logged every second, but it's logged frequently enough that we could compute the missing data by linearly interpolating between consecutive entries.
- Let's consider that at timestamp 10 the plane was at latitude 30 and at timestamp 20, it was at latitude 30.1
- Then we could calculate that at timestamp 16, the plane was at latitude 30.06
- This can be applied for longitude and altitude as well.
- For each query, you will be given the id of a flight and a timestamp. If it happens that the position at that timestamp is known, output that one. Otherwise, find the 2 known values that surround that timestamp and interpolate between them.

> Level 4

	Input	Output
Format	*N* *flightId timestamp (repeats N times)*	*lat long altitude* *(repeats N times)* Repeated for each query in the input. **Validation will be done with a maximum allowed error of 10^{-5}**
Types	**`N (int)`** Number of queries that follow **`flightId (int)`** File name of flight we need to pinpoint **`timestamp (int)`** Timestamp of the moment when we need to locate the flight	**`lat (float)`** North latitude in degrees **`long (float)`** East longitude in degrees **`altitude (float)`** Meters above the sea level
Example	*3* *1469 291102* *760 34054* *4637 207148*	*55.7755 -3.5709999999999997 3703.0* *50.27825 14.953624999999999 11278.0* *50.83725 -0.506875 3595.875*


Good luck!