

LEVEL 5



Kerry Kotl:

Finally, we reach the juicy part, actually transferring the data. There are only a few spots in which we could do this, you know? We're influent enough to halt some planes just so you can find more spots which can be used to transfer data. Don't ask questions, just find me the best way to do it, comrade...

 We'll make it such that planes transmit information from one another when in some degree of proximity. You'll have to find these points of proximity!

 **Task for Level 5:** Calculate the spots where the data can be transmitted from one plane to another 

- › Given a set of flights, find all moments where data can be transferred between them.
- › If it helps, we can **delay flights by a maximum of 1h**.
- › In order to transfer data from flight A to flight B, the **distance between** them has to be **more than 1000m** and **less than a range given** in the test case.
- › Both planes have to be **flying above 6000m**.
- › Transferring data between flights that have the same destination is pointless. **Ignore all cases where A and B land at the same airport**.
- › **You can only delay a flight and not make it take off earlier.**

- › When outputting the transfer points, **consider the delay to be just for the recipient**, since you can add x seconds of delay to both flights and the transfer conditions will still hold, as long as the delay does not exceed 1h.
- › To specify transfer moments, print id of flight A, id of flight B, delay added to flight B, and the transfer timestamps.
- › If you have **multiple consecutive timestamps** where data can be transferred, print only the **first** and the **last** time, with a **dash** ("") in **between**.

	Input	Output
Format	<p><i>transferRange</i> <i>N</i> <i>flightId</i> (repeats <i>N</i> times)</p>	<p><i>flightA</i> <i>flightB</i> <i>delayOfB</i> <i>timePeriods</i></p> <p>The <i>timePeriods</i> will be separated by spaces and can either be:</p> <ul style="list-style-type: none"> • a single integer denoting a timestamp where transfer can happen • 2 integers separated by “-” meaning that for all timestamps in the closed interval, transfer can happen • timestamps will be absolute values • periods will be sorted in ascending order <p>Sort output lines by <i>flightA</i>, then <i>flightB</i> and finally <i>delayOfB</i></p>
Types	<p>transferRange (float) The maximum distance where transfer can happen. In meters N (int) Number of entries that follow flightId (int) File name of flight we need to pinpoint</p>	<p>flightA (string) Id of the flight that transfers the data flightB (string) Id of the flight that receives the data delayOfB (int) Number of seconds that flight B has to wait before taking off timestamp (int) Number of seconds that passed since moment 0</p>
Example	<p>2000.0 3 2866 2878 4424</p>	<p>Complete output is in the archive</p> <p>...</p> <p>4424 2878 1000 223642-223667 223676-223687 4424 2878 1001 223643-223663 223678-223686 4424 2878 1002 223645-223662 223680-223685 4424 2878 1003 223648-223660 223682-223683 4424 2878 1004 223650-223658 4424 2878 1005 223654 ...</p> <p>* Flight 4424 can transfer data to flight 2878 at timestamps 223650, 223651 ... 223657, 223658, if flight B takes off 1004 seconds later than normal.</p>



Good luck!

data by Spire



event organizer

cloudflight