Homework

Problem overview

Humanity has come a long way since your interview! Warp drive has been discovered and people have colonized numerous planets around the galaxy. Apart from the obvious challenges of living in inhospitable environments the Interplanetary Transit Authority (ITA) has been plagued with problems rolling out its travel fare system. They have issued a transit card to every citizen of the Federation. Each card contains:

- Passport ID (unique per person)
- Cash balance (initially 100 florples)
- Timestamped transaction history
- Authenticity hash of id, balance, and history

The ITA controls a network of stations throughout interstellar space.

- A traveler may stay aboard a space station as long as they choose
- A traveler is charged three florples when they leave a station
- A traveler can boost their florple balance at any time by visiting a space station FTM (Florple Teller Machine)
- When florples are debited or credited a station will update the card balance, transaction history and authenticity hash

For example citizen Rick recently departed five space stations, including a sight seeing tour from m87 that returned him to the same station, then finally traveled back to Pluto where he withdrew 7 florples from a FTM. Rick's card now contains the following data:


```
{
   "id":"A83L463F79B3",
   "florples":92,
   "history": [
      { "date":"2019-02-08T12:37:10.00Z", "change":7, "station":"pluto" },
      { "date":"2019-02-06T12:35:10.00Z", "change":-3, "station":"m87" },
      { "date":"2019-02-05T23:71:40.00Z", "change":-3, "station":"m87" },
      { "date":"2019-02-04T09:32:90.00Z", "change":-3, "station":"betelgeuse" },
      { "date":"2019-02-04T08:05:20.00Z", "change":-3, "station":"vega" },
      { "date":"2019-02-03T11:25:30.00Z", "change":-3, "station":"pluto" },
      { "date":"2019-02-01T14:11:12.00Z", "change":100, "station":"pluto" }
   ],
   "hash":"A83K39FNJSL3KSLJFK3JFLXJCLKJEK3"
}
```


Recently the ITA located a freighter in deep space containing a payload of cloned transit cards!

- A cloned card and its real counterpart have the same id; they are *equivalent*
- A cloned card initially has the same florple balance, transaction history, and hash value as the card it was cloned from

To combat fraud each space station now scans the card of every departing citizen to detect cheating. As the head of ITA security your job is to write the detection software that runs on a station.

Details

- Your software runs independently on each space station. Stations do not communicate with each other, so your software only knows about local departures
- You software does not receive notifications when a citizen refills their florple balance, however a refill is recorded in a citizen's card data
- Your software receives a stream of time-sequential departure records from the local station. Each record contains the name of the local station from which the passenger departed and the content of the scanned passenger card (see below)
- Each card contains a hash value that is computed from its id, balance, and history. The hash function will be provided to you, and you can assume that no cheaters have access to it


```
{
   "station":"pluto",
   "passenger":{
      "id":"A83L463F79B3",
      "florples":89,
      "history": [
         { "date":"2019-02-09T05:13:20.00Z", "change":-3, "station":"pluto" },
         { "date":"2019-02-08T12:37:10.00Z", "change":7, "station":"pluto" },
         { "date":"2019-02-06T12:35:10.00Z", "change":-3, "station":"m87" },
         { "date":"2019-02-05T23:71:40.00Z", "change":-3, "station":"m87" },
         { "date":"2019-02-04T09:32:90.00Z", "change":-3, "station":"betelgeuse" },
         { "date":"2019-02-04T08:05:20.00Z", "change":-3, "station":"vega" },
         { "date":"2019-02-03T11:25:30.00Z", "change":-3, "station":"pluto" },
         { "date":"2019-02-01T14:11:12.00Z", "change":100, "station":"pluto" }
      ],
```

```
            "hash":"FK3LSFI39FKFHFL3FGLS93KFJSL49FJSLFKFJ"
        }
    }
```


Types of cheating

Your software should try to catch the following kinds of cheating:

- A card with data that has been tampered with, i.e. forged id, florple balance, history, or hash
- A card that has been duplicated by someone on the local station
- A card that has been duplicated on a different station, and passes through the local station

Cheating example

Here is an example of three sequential departure records on the station vega where equivalent (duplicate) cards are detectable:


```
{
  "station":"vega",
  "passenger":{
    "id":"d54acccc-181b-4b1d-9c55-14213a42af86",
    "florples":73,
    "history":[
      {"date":"2019-07-03T14:40:44.376442Z","change":-3,"station":"vega"},
      {"date":"2019-06-30T14:53:13.376442Z","change":-3,"station":"vega"},
      {"date":"2019-06-29T12:29:23.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-25T14:07:26.376442Z","change":-3,"station":"pluto"},
      {"date":"2019-06-24T03:39:49.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-21T08:45:36.376442Z","change":-3,"station":"betelgeuse"},
      {"date":"2019-06-19T09:38:13.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-16T01:40:23.376442Z","change":-3,"station":"betelgeuse"},
      {"date":"2019-06-14T07:34:39.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-11T20:13:49.376442Z","change":100,"station":"centaurus"}
    ],
    "hash":"Yo6kityHGx7gqntkJqZzd5iq98KocFVDkAij83z3KjA\u003d"
  }
}

{
  "station":"vega",
  "passenger":{
    "id":"d54acccc-181b-4b1d-9c55-14213a42af86",
    "florples":82,
    "history":[
      {"date":"2019-07-07T01:25:29.376442Z","change":-3,"station":"vega"},
      {"date":"2019-07-07T01:20:29.376442Z","change":12,"station":"vega"},
      {"date":"2019-07-03T14:40:44.376442Z","change":-3,"station":"vega"},
      {"date":"2019-06-30T14:53:13.376442Z","change":-3,"station":"vega"},
      {"date":"2019-06-29T12:29:23.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-25T14:07:26.376442Z","change":-3,"station":"pluto"},
      {"date":"2019-06-24T03:39:49.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-21T08:45:36.376442Z","change":-3,"station":"betelgeuse"},
      {"date":"2019-06-19T09:38:13.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-16T01:40:23.376442Z","change":-3,"station":"betelgeuse"},
      {"date":"2019-06-14T07:34:39.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-11T20:13:49.376442Z","change":100,"station":"centaurus"}
    ],"hash":"pGc6kyQz0bgNb3JXRqVhUFFQdt6EwdjBAj+YZbrfr9w\u003d"
  }
}

{
  "station":"vega",
  "passenger":{
    "id":"d54acccc-181b-4b1d-9c55-14213a42af86",
    "florples":73,
    "history":[
      {"date":"2019-07-07T13:40:44.376442Z","change":-3,"station":"vega"},
      {"date":"2019-06-30T14:53:13.376442Z","change":-3,"station":"vega"},
      {"date":"2019-06-29T12:29:23.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-25T14:07:26.376442Z","change":-3,"station":"pluto"},
```

```
      {"date":"2019-06-24T03:39:49.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-21T08:45:36.376442Z","change":-3,"station":"betelgeuse"},
      {"date":"2019-06-19T09:38:13.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-16T01:40:23.376442Z","change":-3,"station":"betelgeuse"},
      {"date":"2019-06-14T07:34:39.376442Z","change":-3,"station":"centaurus"},
      {"date":"2019-06-11T20:13:49.376442Z","change":100,"station":"centaurus"}
    ],
    "hash":"lOElqTErI4pN4tzjw5YhjQi0NcHsu+QUGsGTwUSkGVU\u003d"
  }
}
```


Provided for you

- Five gzipped space station travel log files. Each log file contains all of the departure records for one station that occurred in the month of July, 2019 in time-sequential order
- Java code that compiles with Java 8+ and includes
  - Main class SpaceStationSecurity.java that runs your code and prints the number of violators found to the console
  - Java code that reads the next time-sequential departure record from the local travel log file
  - A function for calculating a card's hash value. You can assume that no cheaters have access to this hash function
  - Gson and Annotations jar dependencies for the helper classes

Your task

- Implement the method SpaceStationSecurity.isViolation(DepartureRecord record). This method should return true if the scanned card has a known equivalent
- When you detect that a card has known equivalent it should be blacklisted; any further departures using the card's passenger ID should be flagged as a violation
- Do not pre-process any log files; rely only on the stream of records passed to isViolation()
- Make an attempt to validate your solution. This part is open ended and you may use any of the logs or even generate your own

Starting conditions

At the beginning of July there are already many citizens on space stations, possibly with cloned transit cards!

Program output

NOTE: It is recommended that you use an IDE to link the included jar dependencies, this will make compiling much easier

> *java SpaceStationSecurity /path/to/ONE/log/file*
>
> *Violations found: 350*
> *(Note: this is not a correct answer)*

Deliverables

- Solution source code
- Give an example for each type of cheating that you try to detect (this helps us understand the intent of your code)
- One paragraph of text describing your approach to validating your solution
- Validation source code

Ask Questions!

It never hurts to ask questions if something needs clarification. Feel free to email us at project@dexterity.capital. We will try to respond promptly to your query!

Evaluation

Your code will be evaluated on the following metrics:

- Correctness of program output
- Code structure and readability
- Validation approach and implementation