Back

logread

`logread` queries the state of the gallery. It prints which employees and guests are in the gallery or its rooms, and allows for various time-based queries of the state of the gallery. The following invocations *must* be supported:

```
logread -K <token> -S <log>
logread -K <token> -R (-E <name> | -G <name>) <log>
```

The following invocations are *optional* (for extra points). If you do not implement an optional feature, be sure to **print `unimplemented` to standard output** when an optional argument is provided.

```
logread -K <token> -T (-E <name> | -G <name>) <log>
logread -K <token> -I (-E <name> | -G <name>) [(-E <name> | -G <name>) ...] <log>
```

As per the above invocations, only one of `-S`, `-R`, `-T`, or `-I` may be specified at once.

In what follows, we refer to employees or visitors who are 'in the gallery'. Each person is expected to first enter the gallery (using `logappend` option `-A`) prior to entering any particular room of the gallery. Once in the gallery, he or she may enter and leave various rooms (using `logappend` options `-A -R` and options `-L -R`, respectively). Finally, the person will leave the gallery (using `logappend` option `-L`). During this whole sequence of events, a person is considered to be 'in the gallery'. See the examples for more information.

When output elements are comma-separated lists, there will be no spaces before or after the commas.

- `-K` *token* Token used to authenticate the log. This token consists of an arbitrary sized string of alphanumeric characters and will be the same between executions of `logappend` and `logread`. If the log cannot be authenticated with the token (i.e., it is not the same token that was used to create the file), then "security error" should be printed to stderr and 255 should be returned.

- `-S` Print the current state of the log to stdout. The state should be printed to stdout on at least two lines, with lines separated by the `\n` (newline) character. The first line should be a comma-separated list of employees currently in the gallery. The second line should be a comma-separated list of guests currently in the gallery. The remaining lines should provide room-by-room information indicating which guest or employee is in which room. Each line should begin with a room ID, printed as a decimal integer, followed by a colon, followed by a space, followed by a comma-separated list of guests and employees. Room IDs should be printed in ascending integer order, all guest/employee names should be printed in ascending lexicographic string order.

- `-R` Give a list of all rooms entered by an employee or guest. Output the list of rooms in chronological order. If this argument is specified, either -E or -G must be specified. The list is printed to stdout in one comma-separated list of room identifiers.

- `-T` Gives the total time spent in the gallery by an employee or guest. If the employee or guest is still in the gallery, print the time spent so far. Output in an integer on a single line. This feature is optional. If the specified employee or guest does not appear in the gallery, then nothing is printed.

- `-I` Prints the rooms, as a comma-separated list of room IDs, that were occupied by all the specified employees and guests at the same time over the complete history of the gallery. Room IDs should be printed in ascending numerical order. This feature is optional. If a specified employee or guest does not appear in the gallery, it is ignored. If no room ever contained all of the specified persons, then nothing is printed.

- `-E` Employee name. May be specified multiple times when used with `-I`.

- `-G` Guest name. May be specified multiple times when used with `-I`.

- `log` The path to the file log used for recording events. The filename may be specified with a string of alphanumeric characters (including underscores).

If `logread` is given an employee or guest name that does not exist in the log, it should print nothing about that employee (which may result in empty output).

If `logread` cannot validate that an entry in the log was created with an invocation of `logappend` using a matching token, then `logread` should return a 255 and print "integrity violation" to `stderr`.

Return values and error conditions

Some examples of conditions that would result in printing "invalid" or "integrity violation" (and return 255):

- `-I` or `-T` used specifying an employee that does not exist in the log, should print nothing and exit with return 0
- If the logfile has been corrupted, the program should exit and print "integrity violation" to stderr and exit with return 255