Exercise #17
- Select maximum temperature for each date
  - Input: two structured textual files containing the temperatures gathered by a set of sensors
    - Each line of the first file has the following format:
      sensorID, date, hour, temperature
    - Each line of the second file has the following format:
      date, hour, temperature, sensorID
  - Output: the maximum temperature for each date (considering the data of both input files)

Exercise #17 - Example
- Input files
  1. 2016-01-01, 14:00, 20.5
  2. 2016-01-02, 14:00, 20.2
  3. 2016-01-01, 14:10, 11.5
  4. 2016-01-02, 14:10, 30.2

- Output
  1. 2016-01-01
    2. 2016-01-02
  3. 30.2
  4. 31.5

Exercise #18
- Filter the readings of a set of sensors based on the value of the measurement
  - Input: a set of textual files containing the temperatures gathered by a set of sensors
    - Each line of the files has the following format:
      sensorID, date, hour, temperature
  - Output:
    - The lines of the input files associated with a temperature value greater than 30.0

Exercise #18 - Example
- Input file
  1. 2016-01-01, 14:00, 20.5
  2. 2016-01-02, 14:00, 30.2
  3. 2016-01-01, 14:10, 11.5
  4. 2016-01-02, 14:10, 30.2

- Output file
  1. 2016-01-02, 14:00, 30.2
  2. 2016-01-02, 14:10, 30.2
**Exercise #19**

- Filter the readings of a set of sensors based on the value of the measurement
  - Input: a set of textual files containing the temperatures gathered by a set of sensors
    - Each line of the files has the following format:
      ```plaintext
      sensorID, date, hour, temperature
      ```
  - Output:
    - The lines of the input files associated with a temperature value less than or equal to 30.0

**Exercise #19 - Example**

- Input file
  ```plaintext
  s1, 2015-01-01, 14:00, 20.5
  s2, 2015-01-01, 14:00, 30.2
  s1, 2015-01-02, 14:10, 11.5
  s2, 2015-01-02, 14:10, 30.2
  ```

- Output file
  ```plaintext
  s1, 2015-01-01, 14:00, 20.5
  s1, 2015-01-02, 14:10, 11.5
  ```

**Exercise #20**

- Split the readings of a set of sensors based on the value of the measurement
  - Input: a set of textual files containing the temperatures gathered by a set of sensors
    - Each line of the files has the following format:
      ```plaintext
      sensorID, date, hour, temperature
      ```
  - Output:
    - A set of files with the prefix "high-temp-" containing the lines of the input files with a temperature value greater than 30.0
    - A set of files with the prefix "normal-temp-" containing the lines of the input files with a temperature value less than or equal to 30.0

**Exercise #20 - Example**

- Input file
  ```plaintext
  s1, 2015-01-01, 14:00, 20.5
  s2, 2015-01-01, 14:00, 30.2
  s1, 2015-01-02, 14:10, 11.5
  s2, 2015-01-02, 14:10, 30.2
  ```

- Output files
  - high-temp-00001
    ```plaintext
    s2, 2015-01-01, 14:00, 30.2
    s2, 2015-01-02, 14:10, 30.2
    ```
  - normal-temp-00001
    ```plaintext
    s1, 2015-01-01, 14:00, 20.5
    s1, 2015-01-02, 14:10, 11.5
    ```