

“Homework” Evaluated Task

General

This task aims to evaluate the relevant skills to write a working application that has WiFi oriented aspects.

Evaluating Criteria

Tasks will be evaluated based on the following criteria:

1. It works as specified.
2. Efficient in memory.
3. Modularized, readable & clean code.

The task

Write an application in your favorite platform (OS and Programming language) that does the following:

1. Upon execution, receives input parameters (by configuration file / cli input / etc...):
 - a. RSSI value (signal strength) threshold (higher than the threshold will be defined as “Inside the store Mac Addresses” and lower/equal will be considered as “Outside of the store Mac Addresses”).
2. As a data input, calls an api (every 1 second to keep getting updated data):
 - a. URL: “<http://api.analoc.com/test/mac-list>”.
 - b. Format: JSON.
 - c. Returns in the Body of the response an array with objects containing data of the 1000 latest up to date detected wifi devices. Response Example:

```
[
  {"mac": "62:A1:51:C5:40:52", "power": -92, "sensor_id": "280", "timestamp": "2014-10-28 08:55:42"},
  {"mac": "54:E6:FC:CE:DF:FC", "power": -83, "sensor_id": "240", "timestamp": "2014-10-28 08:55:41"},
  {"mac": "00:04:20:2C:ED:E2", "power": -85, "sensor_id": "240", "timestamp": "2014-10-28 08:55:40"}
]
```

- “mac” - stands for device MAC address.
- “power” - stands for RSSI value (signal strength: negative, lower is weaker).
- “sensor_id” - the sensor that picked up the signal.
- “timestamp” - when the device was detected.

3. Outputs (to stdout/cli/gui/text file) a refreshing list (refreshes every 1 second):
 - a. Updated number of **distinct** detected WiFi devices grouped by device manufacturer type (Apple/Samsung/LG/...) and split to “Inside the store Mac Addresses” and “Outside of the store Mac Addresses” by the power value from the starting time of the application.
 Output example:

Application Output:
Inside the store:
Apple Inc.: 12 distinct devices
Samsung Ltd.: 7 distinct devices
Unrecognized: 20 distinct devices
Outside of the store:
Apple Inc.: 20 distinct devices
Samsung Ltd.: 13 distinct devices
Unrecognized: 45 distinct devices
 - b. **Note:** The output must display the number of distinct devices from the starting time of the application - So you must take care of potential duplicate devices between api calls.
 - c. **Note:** Attached with this document is a file “oui.txt” that contains manufacturers and their representative initial 6 characters for identifying a the mac’s manufacturer.
4. After 2 minutes stops the api calls, opens a **thread pool** (pool size: 10) and inserts **asynchronously** to a **text file** all of the unique (mac addresses, sensor) pairs that were collected in those 2 minutes so that every thread inserts a group of mac addresses sent by a single sensor_id until there are no more sensors left. (The groups of mac addresses should be separated by a line indicating the sensor ID number).

Good luck!