

Computing Basic

A

School of Computer Science
Universidad de Oviedo

1 The file

There is a file `.ini` in Campus Virtual. It is a text file for setting environment variables.

The structure of this file is very simple. Each line starts in the first column and can be one of the following types: section lines, blank lines, assignment lines or comment lines. No other line is possible.

Section lines Section lines contain a section name in square brackets. The section name can contain just alphabetic characters.

Assignment lines Assignment lines are `key = value` pairs with optional spaces around the `=` sign. The `key` can contain only letters. The `value` can contain letters, digits and whitespaces.

Blank and Comment lines In addition, there can be empty lines (containing just `\t`, whitespaces and new lines) and lines starting with `#`, which are comments and can contain any characters.

There are no repeated names for sections or variables.

For instance, these are the lines of `inifileA.ini`:

```
# inifileA.ini
```

```
# comment  
[alpha]
```

```
base = moon  
ship= alpha 3
```

```
[earth]  
# ?  
base=London  
ship= x-wing
```

```
[beta  
letter=b  
variable = hola
```

2 Get the content of the file (1p)

Write a function `format_file(F)` that receives one parameters (string) corresponding to a filename. This function returns a list containing formatted lines in `F` in the way:

- Don't return blank lines
- Don't return comment lines
- Get rid of newlines at the end of the lines

Example: `format_file('inifileA.ini')` will generate a list (items are in different lines because of space limitations):

```
['[alpha]', 'base = moon', 'ship= alpha 3',  
'[earth]', 'base=London', 'ship= x-wing'],  
'[beta]', 'letter=b', 'variable = hola']
```

When formatted, the final list will contain just sections and assignments.

3 Check wrong lines (3p)

Write a function `find_errors(L)` that receives a list `L` generated by `format_file` and returns a list containing the numbers of lines containing errors. A line contains an error if

- the first character is `[` and there is no `]` at the end of the line
- the first character is `[`, the last is `]` and the string in the middle doesn't match the section name format.

You can count lines starting at 0. **Example:** `print find_errors(format_file('inifileA.ini'))` would print `[6]`

4 Generate an error-free file (3p)

Write a function `error_free(F)` that receives a filename and generates a new file, with the original name plus `ErrorFree` and the same extension. This new file is a **formatted**, error-free copy of `F`.

Example: `errors_free('inifileA.ini')` will generate a file named `inifileAErrorFree.ini` with the following content:

```
[alpha]  
base = moon  
ship= alpha 3  
[earth]  
base=London  
ship= x-wing  
letter=b  
variable = hola
```

5 statistics (3p)

Write a function `statistics(F)` that receives an error-free file filename (`F`) and prints out some lines with key-value pairs separated by `=`. The possible pairs are:

- `total sections` plus the number of sections correctly defined in filename
- `section name` plus the number of variables defined in this section

Example: `statistics('inifileAErrorFree.ini')` will print:

```
alfa=2
earth=4
total sections=2
```

6 When finish...

Write docstrings in all the functions.

Identify the author in the first lines.

Save your python file named **test3A.py**.

Upload this only file to the corresponding task in Campus Virtual.