CS350 OPERATING SYSTEMS COURSE
TERM PROJECT REPORT

DATE: 01/06/2018
NAME: Ercument Burak Tokman – S013090
TOPIC: Analyze file count, size and type information on different operating systems with Python script

REQUIRED SOFTWARE

- Python 3.6 - Conda 4.4.11 - Sublime Text 3 Build 3143

REQUIRED PYTHON LIBRARIES

- **Standard:** os - os.path - sys - Subprocess - Optparse - Pathlib
- **External:** termcolor - python-magic - python-magic-bin

0. ABSTRACT

Python script that analyzes file size, type information and file count, size in folder on different operating systems. Script testing done on Windows 10 and macOS systems. Both forking fine, haven't done the testing but it will most likely work on Linux and more other operating systems.
Accomplishing the aim of this project be done with standard & external python modules, system calls and executing command on the system. That's why the output of the script separated into two parts. Part 1: Python Modules and Part 2: Commands & System Calls.

1. BACKGROUND RESEARCH

Total time done on the background research no more than a week. Finding out the right command combination, pipelining and executing this command with python took the most of the time. Passing parameters to python script and recording the output of executed commands also took much time. Reading documents of the required libraries done quickly.

1.1 PYTHON

Interpreted high-level programming language for general purpose programming. Python has steady learning curve because of its ease to use. Also highly used in big data, machine learning and statistics. It has chosen as programming language of this project.

1.2 SUBLIME TEXT

Very simple text editor. Because most of the time using IDE for Python does not needed, it's possible to make this project completely using a text editor.

1.3 CONDA

Conda provides package manager and environment management for Python. It's used at the beginning of this project yet it's completed without further need for it. Python's built in package manager **pip** comes with Python installation and for the external modules that needed in this project, they do not come with **Conda**. So it's become pointless to use **Conda** anymore.

1.4 OS MODULE

OS module in Python provides a way of using operating system dependent functionality. The functions that the OS module provides allows you to interface with the underlying operating system that Python is running on Windows, Mac or Linux. **os.stat** and **os.system** functions will be used in this project. **os.stat** gets the status of a file. Performs system call that equivalent to 'stat()

1.5 SUBPROCESS MODULE

OS module has **os.system()** function and it has the ability to make system call, but its deprecated since Python version 2.6 so we will be using subprocess module in this case that allows us to create new processes and connect their I/O's through pipe and return output.

1.6 SYS MODULE

Interpreter need access to use or maintain some variables, functions that can interact and **sys** module makes this possible. Sys module used to make python script able to take arguments which is file or folder path in this case.

1.7 OPTPARSE MODULE

Python script requires a file or folder link to be provided as parameter then script can be fully initiated. Optparse module makes it possible to python script to receive parameters.

1.8 PATHLIB MODULE

Object-oriented filesystem paths module. Makes it easy to work with directories on different operating systems.

1.9 TERMCOLOR MODULE

Print colored outputs on terminal. Perfectly working on Linux and macOS. Not so perfect with Window's Command Prompt, not always displaying perfectly. Yet I've used it because it makes the output much more readable.

1.10 PYTHON-MAGIC MODULE

Python-magic is a python interface to the libmagic file type identification library. libmagic identifies file types by checking their headers according to a predefined list of file types. This functionality is exposed to the command line by the Unix command **file**.

1.11 PYTHON-MAGIC-BIN MODULE

Module uses ctypes to access the libmagic file type identification library. It makes use of the local magic database and supports both textual and MIME-type output.

2. INSTALLATION

2.1 PYTHON

Download Python 3.6.5 installation package from it's official website depending to your operating system. Required link provided in references.

2.2 MODULES

Two different external modules used in this project that can be installed via pip (Python Package Manager) or linked to project manually.

Type the command below to install required modules via pip (before executing the command you have to go to scripts directory in terminal):

```
pip install -r requirements.txt
```

2.3 CONDA

Download and install **Conda** from it's official website. Yet it is not needed for this project anymore. But it gives almost all of the most used Python libraries so if you are thinking to use Python more in future than it is a good choice.

2.4 SUBLIME TEXT

Download & install from official website. Link provided in **references** part.
Installation does not require anything special.

3. SCRIPT STRUCTURE

Script begins with detecting the operating system type. Output will be Windows, macOS or Linux depending to the system. Then check if any parameter provided with option **-f**. Argument must be file or folder path. Script do checking on that path. If argument is file than do **file type analysis** on that file. Output will show size and type information of the file. If folder path given as argument then do **folder type analysis.** Output will show file and folder count in that directory and it will list all of the files in it.

3.1 ANALYSIS

Brief explanation of the code that i used.
Part 1: Python functions are used to do analysis on file.

File size:

```
statinfo = os.stat(filename)
statinfo.st_size
```

File type:

```
magic.from_file(filename)
```

File and folder count:

```
for name in os.listdir(directory):
    ...
```

Part 2: System commands are used to determine file type and size.

File size:

```
file <absolute file path>
```

File type:

```
wc -c <absolute file path>
```

3.2 FOLDER ANALYSIS

Part 1: Do folder analysis with python functions.

List and count files:

```
        os.listdir(directory)
        if os.path.isdir(directory + '/' + name):
                dir_count += 1
        if os.path.isfile(directory + '/' + name):
                file_count += 1
```

Part 2: Execute commands or system calls to do folder analysis.

File and folder count:

```
        folder_count = os.system('cd "' + directory + '" && ls -l |
grep ^d | wc -l')
        file_count = int(os.system('cd "' + directory + '" && ls -l |
grep ^- | wc -l'))
```

4. HOW TO USE

You must complete the installation with all the parts (as I mentioned above) before executing the script, if you don't you will face with some error most likely.
Before using the script I have to remind you that you have to go to script's directory to execute like the commands as shown below or use the **absolute path** of the script then give file or folder as with parameter **-f** in terminal.

```
        python analyze.py -f <file or folder path>
```

i.e. for a **file**:

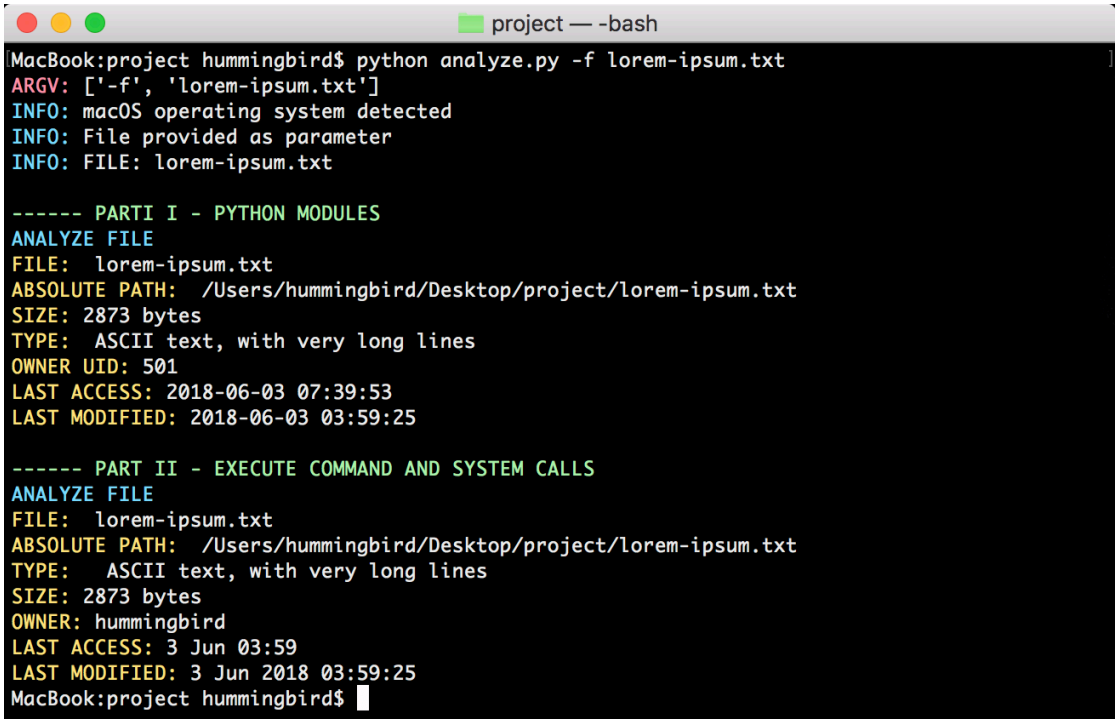
```
        python analyze.py -f /Users/Guest-user/Desktop/lorem-ipsum.txt
```

i.e. for a **folder**:

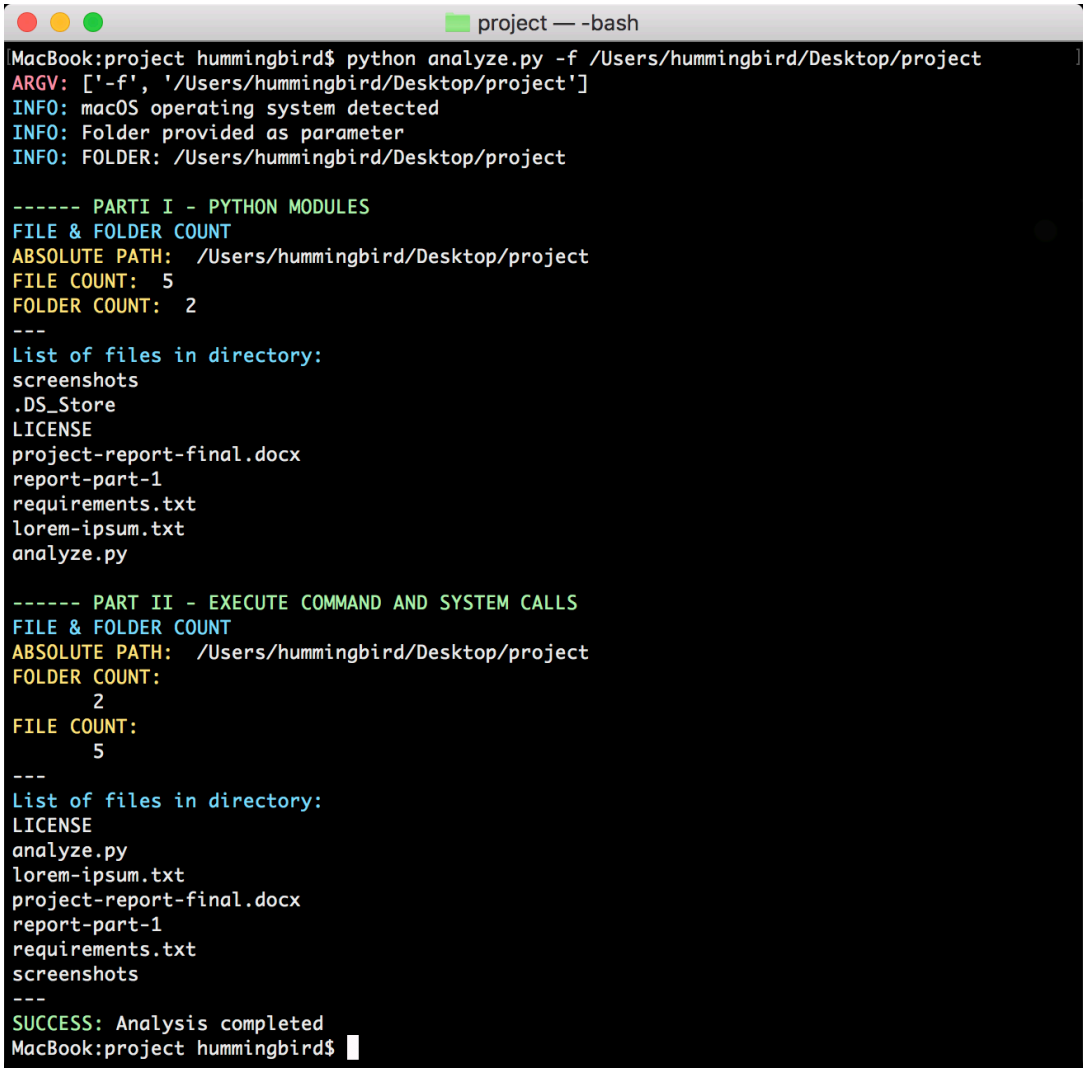
```
        python analyze.py -f /Users/Guest-user/Desktop/project
```

5. TESTING & OUTPUT

Sample output when file provided as parameter:



Sample output when folder provided as parameter:



6. LICENSE

MIT License
Even though it's a term project I've added MIT License. You are free to distribute, modify and use it commercially.

7. REFERENCES

Python - https://www.python.org/downloads/
Conda - https://conda.io/docs/user-guide/install/download.html
Sublime Text - https://www.sublimetext.com/
python-magic - https://github.com/ahupp/python-magic
termcolor – https://pypi.org/project/termcolor/
sys - https://docs.python.org/3/library/sys.html
Optparse - https://www.saltycrane.com/blog/2009/09/python-optparse-example/
The MIT License - https://opensource.org/licenses/MIT
Windows Shell List Files - https://stackoverflow.com/questions/23228983/batch-file-list-files-in-directory-only-filenames
Windows Shell File Count - https://serverfault.com/questions/110725/windows-command-prompt-how-to-get-the-count-of-all-files-in-current-directory