CMPE230 Homework 2

Burak Yüksel – Fahri Can Şanlı

2015400225 – 2015400096

Problem Statement

In this project, we are asked to implement a file utility program called filelist that will traverse the given directories recursively and report path names of files that satisfy some search criteria in Python.

There is list of options to restrict the search operation and to get different type of outputs.

Solution Description

First, we parsed the arguments by using argparse module and we defined all possible options. Then, we rearranged the inputs of before and after options by adding appropriate strings and input of bigger and smaller options by converting them to bytes for convenience.

After that, we created a deque to traverse directories recursively. At each step we took an element from deque and iterated through its content. Directories inside this element are pushed into deque while files are checked if they satisfy the given conditions if so they are stored in a list.

Finally, by using the list that consist of files satisfying conditions, we carried out some operations according to given set of options which can be delete, zip, duplcont, duplname, stats, nofilelist.

Functions

correct\_argument\_inputs():

Edits argument inputs for convenience.

parse\_arguments():

Creates a parser which accepts options: before, after, match, bigger, smaller, delete, zip, duplcont, duplcont, stats, nofilelist, and a directory list.

before(file):

Gets the last modification time of file. Converts it to the given format YYYYMMDDTHHMMSS and compares them character by character starting from the beginning. Returns whether the file’s last modification time is older than the given time.

after(file):

Gets the last modification time of file. Converts it to the given format YYYYMMDDTHHMMSS and compares them character by character starting from the beginning. Returns whether the file’s last modification time is newer than the given time.

bigger(file):

Gets the size of the file and compares with the given size. Returns true if the size of file is greater than the given size, false otherwise.

smaller(file):

Gets the size of the file and compares with the given size. Returns true if the size of file is less than the given size, false otherwise.

match(name):

Checks whether name matches with given regular expression.

delete(file):

Deletes the file by running a bash command.

zip():

First creates a directory named \_files\_to\_be\_zipped in current directory and copies all the files that satisfy the given conditions to this directory by changing the name if needed. Then, formed a single zip command that contains all the files inside this directory and run it. After zipping, it removes the directory and its contents.

How to Compile

python3 filelist.py [options] [directory list]