

ASSIGNMENT REPORT 1: PROCESS AND THREAD IMPLEMENTATION

CENG2034, OPERATING SYSTEMS

Emre Taşkın
emretaskin2@posta.mu.edu.tr
170709060
Github username : emretask1n

Sunday 10th May, 2020

Abstract

In this lab,I learned how operating systems manage their memories. I learned some python libraries to use linux for systemcalls and processes.

I looked at how the threads and interpreters work in unit time. I looked advantages and disadvantages of multiprocessing and multithreading.

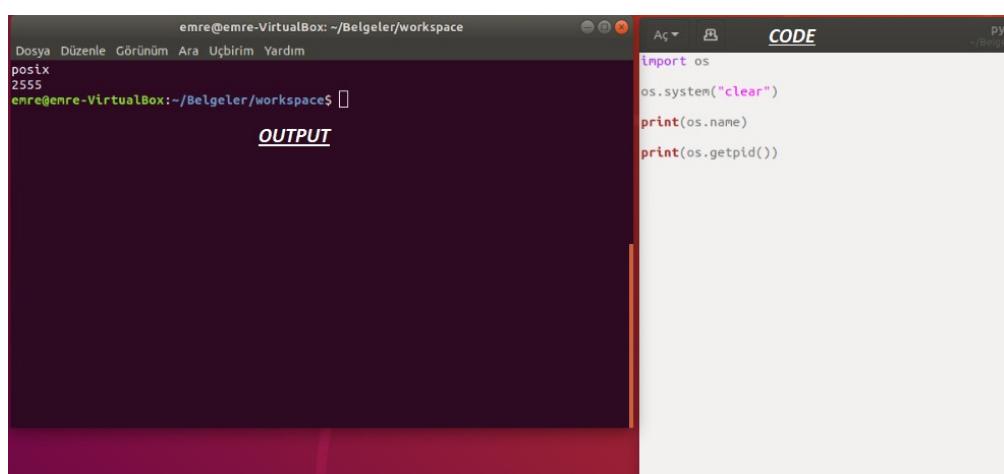
1 Introduction

This labs aim was learning operating systems and their system managings, to have knowledge about operating systems, processes and threads.

2 Assignments

2.1 Assignment 1

First I wrote a python script that writes PID of itself.



The screenshot shows a terminal window titled "emre@emre-VirtualBox: ~/Belgeler/workspace". The terminal has two panes: "CODE" on the right and "OUTPUT" on the left. In the "CODE" pane, there is a single line of Python code: "print(os.getpid())". In the "OUTPUT" pane, the output of the command is shown: "2555".

```
import os
os.system("clear")
print(os.name)
print(os.getpid())
```

2.2 Assignment 2

Second I printed the loadavg of my linux operating system.

The screenshot shows a terminal window with two panes. The left pane is labeled "OUTPUT" and displays the following text:

```
emre@emre-VirtualBox: ~/Belgeler/workspace
Dosya Düzenle Görünüm Ara Uçbirim Yardım
posix
2656
(0.06, 0.2, 0.14)
emre@emre-VirtualBox:~/Belgeler/workspace$
```

The right pane is labeled "CODE" and contains the following Python script:

```
import os
os.system("clear")
print(os.name)
print(os.getpid())
print(os.getloadavg())
```

2.3 Assignment 3

I took and printed “5 min loadavg” value and cpu core count.

The screenshot shows a terminal window with two panes. The left pane is labeled "OUTPUT" and displays the following text:

```
emre@emre-VirtualBox: ~/Belgeler/workspace
Dosya Düzenle Görünüm Ara Uçbirim Yardım
posix
2717
(0.03, 0.06, 0.08)
Load average of last 5 minute is: 0.06
Cpu core count is: 2
emre@emre-VirtualBox:~/Belgeler/workspace$
```

The right pane is labeled "CODE" and contains the following Python script:

```
import os
os.system("clear")
print(os.name)
print(os.getpid())
print(os.getloadavg())
load1 , loads , load15 = os.getloadavg()
print("Load average of last 5 minute is: ", loads)
CPUs = os.cpu_count()
print("Cpu core count is: ", CPUs)
```

2.4 Assignment 4

I checked some links, are they working or not working. First I create a function and determinated its time without using multithreading. Then I used multithreading and looked its time. I saw that multithreading is faster way to check links.

Without using multithreading it took 1.7s

```

Dosya Düzenle Görünüm Ara Uçbirim Yardım
Dosya Düzenle Görünüm Ara Uçbirim Yardım
postx
2865
(0.01, 0.03, 0.01)
Load average of last 5 minute is: 0.03
Cpu core count is: 2
https://api.github.com is a valid url
https://www.python.org is a valid url
http://bilgisayar.mu.edu.tr is a valid url
http://akrepnalan.com/ceng2034 is not valid
https://github.com/caesarsalad-wow is not valid
real    0m1.752s
user    0m0.226s
sys     0m0.047s
emre@emre-VirtualBox:~/Belgeler/workspace$ 
```

OUTPUT

```

import os
import requests

os.system("clear")
print(os.name)
print(os.getpid())
print(os.getloadavg())

load1 , load5 , load15 = os.getloadavg()
print("Load average of last 5 minute is: ", load5)

CPUs = os.cpu_count()

print("Cpu core count is: ", CPUs)

def url_check(url):
    r = requests.head(url)
    if r.status_code == 200:
        print(url , " is a valid url")
    else:
        print(url , "is not valid")

url_check("https://api.github.com")
url_check("https://www.python.org")
url_check("http://bilgisayar.mu.edu.tr")
url_check("http://akrepnalan.com/ceng2034")
url_check("https://github.com/caesarsalad-wow") 
```

CODE

Python ▾ Etiket


```

Dosya Düzenle Görünüm Ara Uçbirim Yardım
Dosya Düzenle Görünüm Ara Uçbirim Yardım
postx
3033
(0.24, 0.21, 0.09)
Load average of last 5 minute is: 0.21
Cpu core count is: 2
http://akrepnalan.com/ceng2034 is not valid
https://bilgisayar.mu.edu.tr is a valid url
https://api.github.com is a valid url
https://www.python.org is a valid url
https://github.com/caesarsalad-wow is not valid
real    0m0.821s
user    0m0.253s
sys     0m0.038s
emre@emre-VirtualBox:~/Belgeler/workspace$ 
```

OUTPUT

With using multi threading it took just 0.8seconds

```

import os
import requests

print(os.getcpu())
print(os.getloadavg())
load1 , loads , load15 = os.getloadavg()
print("Load average of last 5 minute is: ", load5)

CPUs = os.cpu_count()

print("Cpu core count is: ", CPUs)

def url_check(url):
    r = requests.head(url)
    if r.status_code == 200:
        print(url , " is a valid url")
    else:
        print(url , "is not valid")

url_check("https://api.github.com")
url_check("https://www.python.org")
url_check("http://bilgisayar.mu.edu.tr")
url_check("http://akrepnalan.com/ceng2034")
url_check("https://github.com/caesarsalad-wow")

threads=threading.Thread(target=url_check, args=( "https://api.github.com",))
threads2=threading.Thread(target=url_check, args=( "https://www.python.org",))
threads3=threading.Thread(target=url_check, args=( "http://bilgisayar.mu.edu.tr",))
threads4=threading.Thread(target=url_check, args=( "http://akrepnalan.com/ceng2034",))
threads5=threading.Thread(target=url_check, args=( "https://github.com/caesarsalad-wow",))

threads.start()
threads2.start()
threads3.start()
threads4.start()
threads5.start() 
```

Python ▾ Etiket Genişliği: 8 ▾ Sat 44, 50:16 ▾ ARY

3 Results

Python's os library helps to process with codes like os.getpid() provides functions for interacting with the operating system, os.getloadavg() is used to get the load average over the last 1, 5, and 15 minutes, os.cpu_count() returns the number of processors number of cores present in the system. Requests library is simple HTTP library for Python, built for human beings. Multiprocessing and threading can be really useful in some situations. If we use multithreading in a cpu using program that increases the execution time of the process.

4 Conclusion

In this lab, I learned about how linux operating system manages memory and what is process in linux.I ran some process codes from python scripts.I learned and implemented python libraries that I had never used before. I made multiprocessing and multithreading examples. I also learned how can I set up an virtual machine to use linux thanks to this lab.