ASSIGNMENT REPORT 1: PROCESS AND THREAD IMPLEMENTATION

CENG2034, OPERATING SYSTEMS

Gizem Pesen
gizempesen@posta.mu.edu.tr
https://github.com/GPS199

Saturday 2$^{\text{nd}}$ May, 2020

Abstract

This report is for the lab course in Ceng3544 at MSKU University. The main purpose of the lab is to be knowledgeable about thread implementation and use it in daily lives.

1 Introduction

This lab is performed because of the importance of multiprocessing and threads. Specifically the purpose of the lab is to recognize advantages of threads like minimizing the context switching time etc. In addition to them it is repeated GNU/Linux fundamentals installing and using basic scripts.

2 Assignments

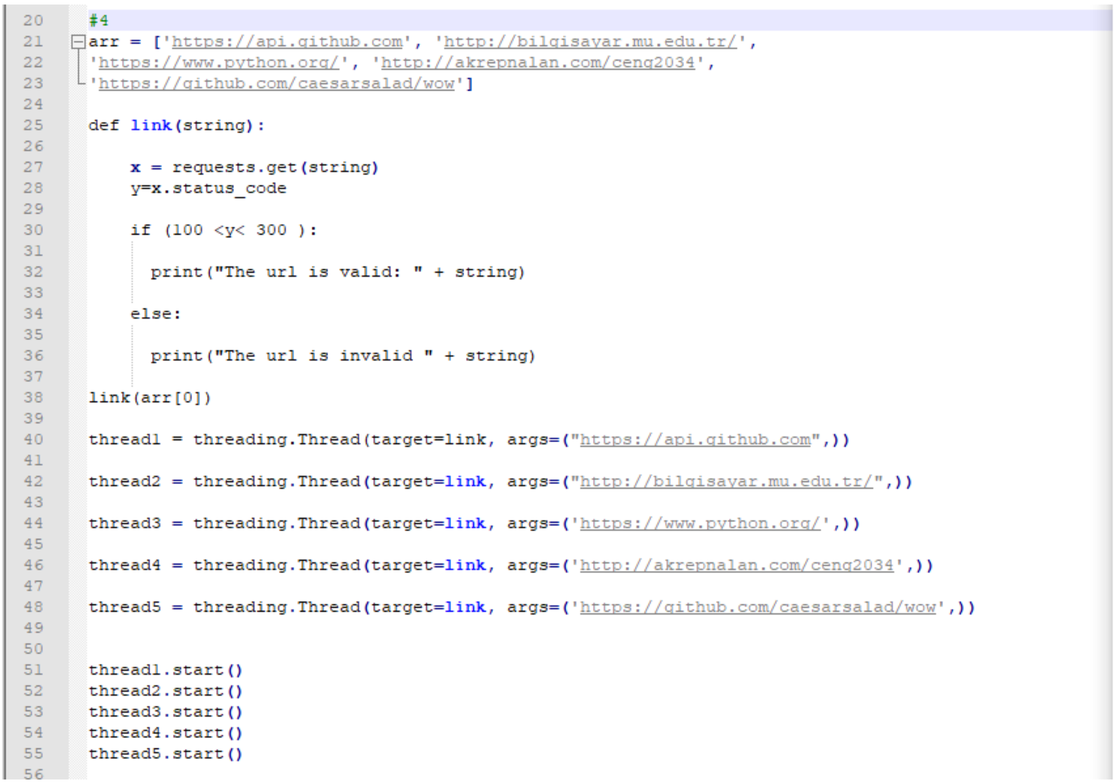
It is worked on https://repl.it/ to try code and fix errors that watching on the lab online lesson. It was easy to access First It is imported os, sys libraries for first three question. In addition, it is imported threading and requests to run the last question that wants to check links valid or invalid.

2.1 Assignment X.x ("Hello world")

The point of the this Assignments section is to show the readers source code of my project. It is prefered to show it with screenshots.Also, there is just the checking links part and it will be explained whole code step by step. *in subfigure*.

Note

In the picture below , I faced with some errors for instance; in lines between 21-23 , first I put the wrong quotation marks. in line 25 , I searched to realize string data structer. in line 30, first I tried the y between 200 and 300 and I get wrong output like whole links are valid. as a result ; I fix these type error and create the code in figure 1

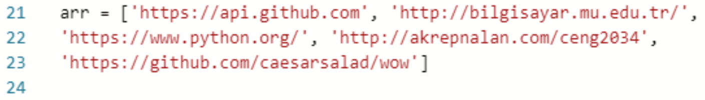


```
#4
arr = ['https://api.github.com', 'http://bilgisayar.mu.edu.tr/',
'https://www.python.org/', 'http://akrepnalan.com/ceng2034',
'https://github.com/caesarsalad/wow']

def link(string):

    x = requests.get(string)
    y=x.status_code

    if (100 <y< 300 ):

      print("The url is valid: " + string)

    else:

      print("The url is invalid " + string)

link(arr[0])

thread1 = threading.Thread(target=link, args=("https://api.github.com",))

thread2 = threading.Thread(target=link, args=("http://bilgisayar.mu.edu.tr/",))

thread3 = threading.Thread(target=link, args=('https://www.python.org/',))

thread4 = threading.Thread(target=link, args=('http://akrepnalan.com/ceng2034',))

thread5 = threading.Thread(target=link, args=('https://github.com/caesarsalad/wow',))


thread1.start()
thread2.start()
thread3.start()
thread4.start()
thread5.start()
```

Figure 1: Screenshot of the code.
( )

3 Results

- repeated GNU/Linux fundamentals
- installing Vmware tools and experienced virtual machine to be familiar in linux world.
- used basic scripts.
- it is worked on terminal and tried threads with lab sections.
- it is experienced process operations and multiprocessing.
- repeated python and used libraries.
- learned checking links with if-else conditions, and print them valid or invalid.
- it is used git for this project , commit and push it with a private reposity.
- it is used latex platform and I write this report in open source writter editor.
- All of the code explained with screenshots on the end of the report.



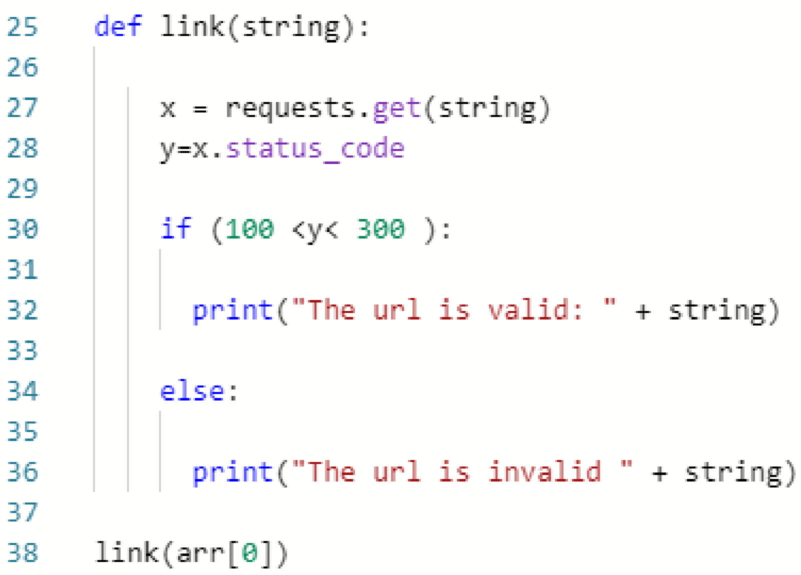
```
arr = ['https://api.github.com', 'http://bilgisayar.mu.edu.tr/',
'https://www.python.org/', 'http://akrepnalan.com/ceng2034',
'https://github.com/caesarsalad/wow']
```

Figure 2: This is the taking array part.
First for checking valid (working) or invalid url , it is created an array that includes 5 links like you see in the given picture.

4 Conclusion

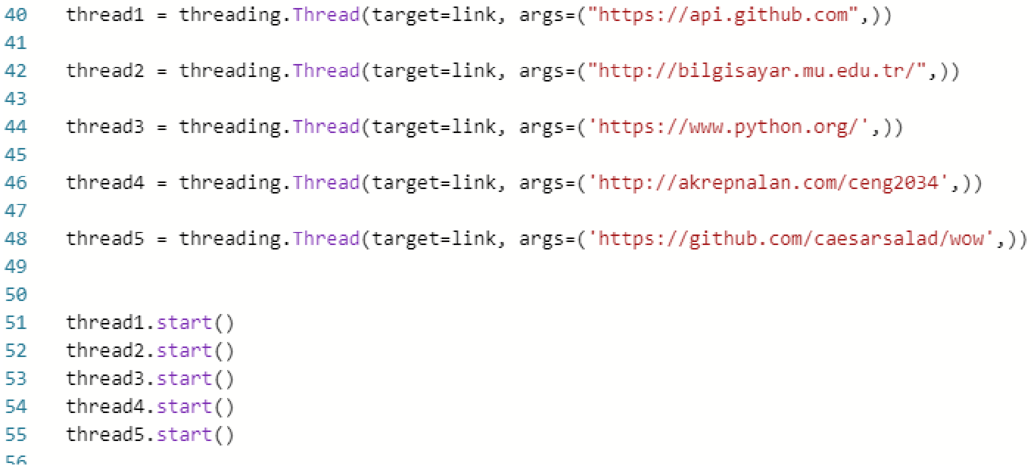
In conclusion , It is learned that specially multiprocessing ,threads ,sending a python to github and latex. Multiprocessing will be easier our works in our future projects. In my opinion , except of the threads and multiprocessing ; this project made me experienced in git and latex.

Git is the hearth of the computer science and I can use it easily thanks to this project, I believe that latex is the one of the useful platforms although first it was hard for me. Sum up, this lab and midterm project made me familiar such good platforms and thread implementation subject.



```python
def link(string):

    x = requests.get(string)
    y=x.status_code

    if (100 <y< 300 ):

        print("The url is valid: " + string)

    else:

        print("The url is invalid " + string)

link(arr[0])
```

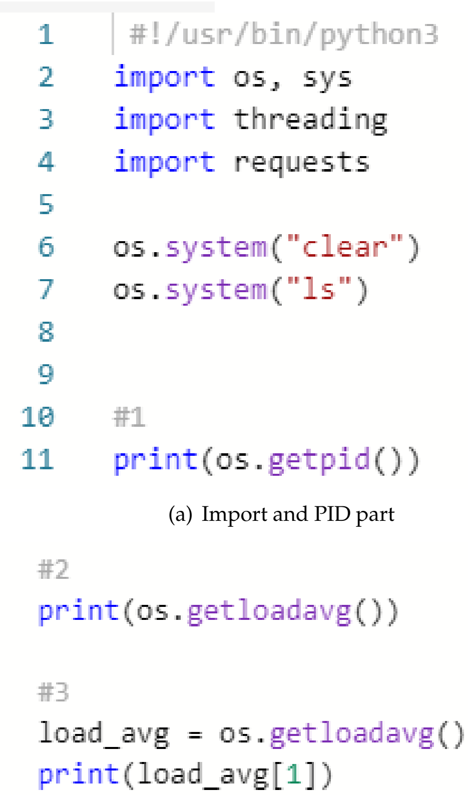
(a) Definition link with string parameter



```python
thread1 = threading.Thread(target=link, args=("https://api.github.com",))

thread2 = threading.Thread(target=link, args=("http://bilgisayar.mu.edu.tr/",))

thread3 = threading.Thread(target=link, args=('https://www.python.org/',))

thread4 = threading.Thread(target=link, args=('http://akrepnalan.com/ceng2034',))

thread5 = threading.Thread(target=link, args=('https://github.com/caesarsalad/wow',))


thread1.start()
thread2.start()
thread3.start()
thread4.start()
thread5.start()
```

(b) Thread part

Figure 3: In definition link picture , it is showing our code body .(a) and the code continue with threading part (right). Notice first it is used string , and we get the links with requests library that we imported *imports are in another screenshot* use of n **??** . .
()



```
#!/usr/bin/python3
import os, sys
import threading
import requests

os.system("clear")
os.system("ls")


#1
print(os.getpid())
```

(a) Import and PID part

```
#2
print(os.getloadavg())

#3
load_avg = os.getloadavg()
print(load_avg[1])
```

(b) loadavg part

Figure 4: Result from fig-a showing Import and PID part (left) and it is showing getting loadavg and print it (right).In first picture it is clear the used libraries that os, sys , threading and requests. (request lib is used for http links.) which was achieved through the use of Equation **??** .
(this is the example)