

LAB 6

Concurrent Server - High-level Network programming



Họ tên và MSSV: Cao Đức An - B1910027

Nhóm học phần: CT293 - 03

1. Bài 1

Cập nhật phần Server của Bài 2-Lab04 và Bài 4-Lab04 để có thể phục vụ song song nhiều Client cùng một lúc (sử dụng kỹ thuật Multi-Threading).

(Chụp hình minh họa code bài làm và kết quả thực thi; đính kèm tập tin code khi nộp bài.)

- Code bài 2 lab 5 server:
- Ta thêm 1 class ClientThread như sau:

```
3 class ClientThread(Thread):
4     def __init__(self, ip, port, sock):
5         Thread.__init__(self)
6         self.ip = ip
7         self.port = port
8         self.sock = sock
9         print(" New thread started for "+ip+": "+str(port))
10    def convert(self, number):
11        if(number.isdigit()):
12            switcher = {
13                0: 'Không',
14                1: 'Một',
15                2: 'Hai',
16                3: 'Ba',
17                4: 'Bốn',
18                5: 'Năm',
19                6: 'Sáu',
20                7: 'Bảy',
21                8: 'Tám',
22                9: 'Chín',
23            }
24            return switcher.get(int(number))
25        else:
26            return "Không phải là số"
27    def run(self):
28        while True: # Receive the data in small chunks and retransmit it
29            data = self.sock.recv(64).decode()
30            if (data):
31                data = self.convert(data);
32                print('sending data back to the client', data)
33                self.sock.send(data.encode())
34            else:
35                print('no data from', ip)
36            break
```

- Sau đó ta hiệu chỉnh chương trình chính để chạy như sau

```
40  host = "localhost"
41  port = 8888
42  tcpsock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
43  tcpsock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
44  tcpsock.bind((host, port))
45  threads = []
46
47  while True:
48      tcpsock.listen(5)
49      print("Waiting for incoming connections...")
50      (conn, (ip, port)) = tcpsock.accept()
51      print('Got connection from ', (ip, port))
52      newthread = ClientThread(ip, port, conn)
53      newthread.start()
54      threads.append(newthread)
```

-
- Kết quả thực thi khi server có nhiều client kết nối tới

```
PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py .\cau2_lab5_server.py
Waiting for incoming connections...
Got connection from ('127.0.0.1', 52590)
New thread started for 127.0.0.1:52590
Waiting for incoming connections...
Got connection from ('127.0.0.1', 52591)
New thread started for 127.0.0.1:52591
Waiting for incoming connections...
Got connection from ('127.0.0.1', 52592)
New thread started for 127.0.0.1:52592
Waiting for incoming connections...
sending data back to the client Bảy
no data from 127.0.0.1
sending data back to the client Năm
no data from 127.0.0.1
sending data back to the client Ba
no data from 127.0.0.1
```

-
- Code bài 4 lab 5 server:
- Ta thêm 1 class ClientThread


```
35         # delete
36         elif(method == "DELETE"):
37             file_path = './' + name
38             try:
39                 os.remove(file_path)
40                 self.sock.send("OK\n".encode())
41                 print("da xoa file ", name, " thanh cong!!")
42             except OSError as e:
43                 self.sock.send("ERROR\n".encode())
44                 print("Error: %s : %s" % (file_path, e.strerror))
45         # list
46         elif(method == "LIST"):
47             file_path = './' + name
48             try:
49                 list = os.listdir(file_path)
50                 self.sock.send("OK\n".encode())
51                 c2, addr2 = tcpsockSendData.accept()
52                 myString = "-".join(list)
53                 c2.send(myString.encode())
54                 print("hien ds trong thu muc ", name, " thanh cong!!")
55                 c2.close();
56             finally:
57                 self.sock.send("ERROR\n".encode())
```

- Sau đó ta hiệu chỉnh chương trình chính:

```
59  host = "localhost"
60  port = 8000
61  port2 = 8001
62  tcpsockSendData = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
63  tcpsockSendData.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
64  tcpsockSendData.bind((host, port2))
65
66  tcpsock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
67  tcpsock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
68  tcpsock.bind((host, port))
69
70  threads = []
71  buffer_size = 1024
72
73  while True:
74      tcpsock.listen(5)
75      tcpsockSendData.listen(5)
76      print("Waiting for incoming connections...")
77      (conn, (ip, port)) = tcpsock.accept()
78      print('Got connection from ', (ip, port))
79      newthread = ClientThread(ip, port, conn)
80      newthread.start()
81      threads.append(newthread)
```

- Kết quả thực thi khi server có nhiều client kết nối tới

```
PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py .\Cau4_lab5_server.py
Waiting for incoming connections...
Got connection from ('127.0.0.1', 52777)
New thread started for 127.0.0.1:52777
Waiting for incoming connections...
Got connection from ('127.0.0.1', 52780)
New thread started for 127.0.0.1:52780
Waiting for incoming connections...
Got connection from ('127.0.0.1', 52783)
New thread started for 127.0.0.1:52783
Waiting for incoming connections...
Exception in thread Thread-3:
Traceback (most recent call last):
  File "C:\Program Files\Python310\lib\threading.py", line 1009, in _bootstrap_inner
    self.run()
  File "D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6\Cau4_lab5_server.py", line 13, in run
    req = self.sock.recv(64).decode()
ConnectionResetError: [WinError 10054] An existing connection was forcibly closed by the remote host
hien ds trong thu muc cau4 thanh cong!!
Exception in thread Thread-1:
Traceback (most recent call last):
  File "C:\Program Files\Python310\lib\threading.py", line 1009, in _bootstrap_inner
    self.run()
  File "D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6\Cau4_lab5_server.py", line 13, in run
    req = self.sock.recv(64).decode()
ConnectionResetError: [WinError 10054] An existing connection was forcibly closed by the remote host
da xoa file 1651808499_data1.txt thanh cong!!
```

2. Bài 2

Tham khảo ví dụ MessageServer.py và MessageClient.py, viết chương trình Chat đơn giản sử dụng UDP socket cho phép hai người trên hai máy tính trò chuyện với nhau. Lưu ý: tạo 2 thread (1 dùng để gửi, 1 để nhận thông điệp) để chương trình cho phép người dùng nhận và gửi thông điệp song song.

(Chụp hình minh họa code bài làm và kết quả thực thi; đính kèm tập tin code khi nộp bài.)

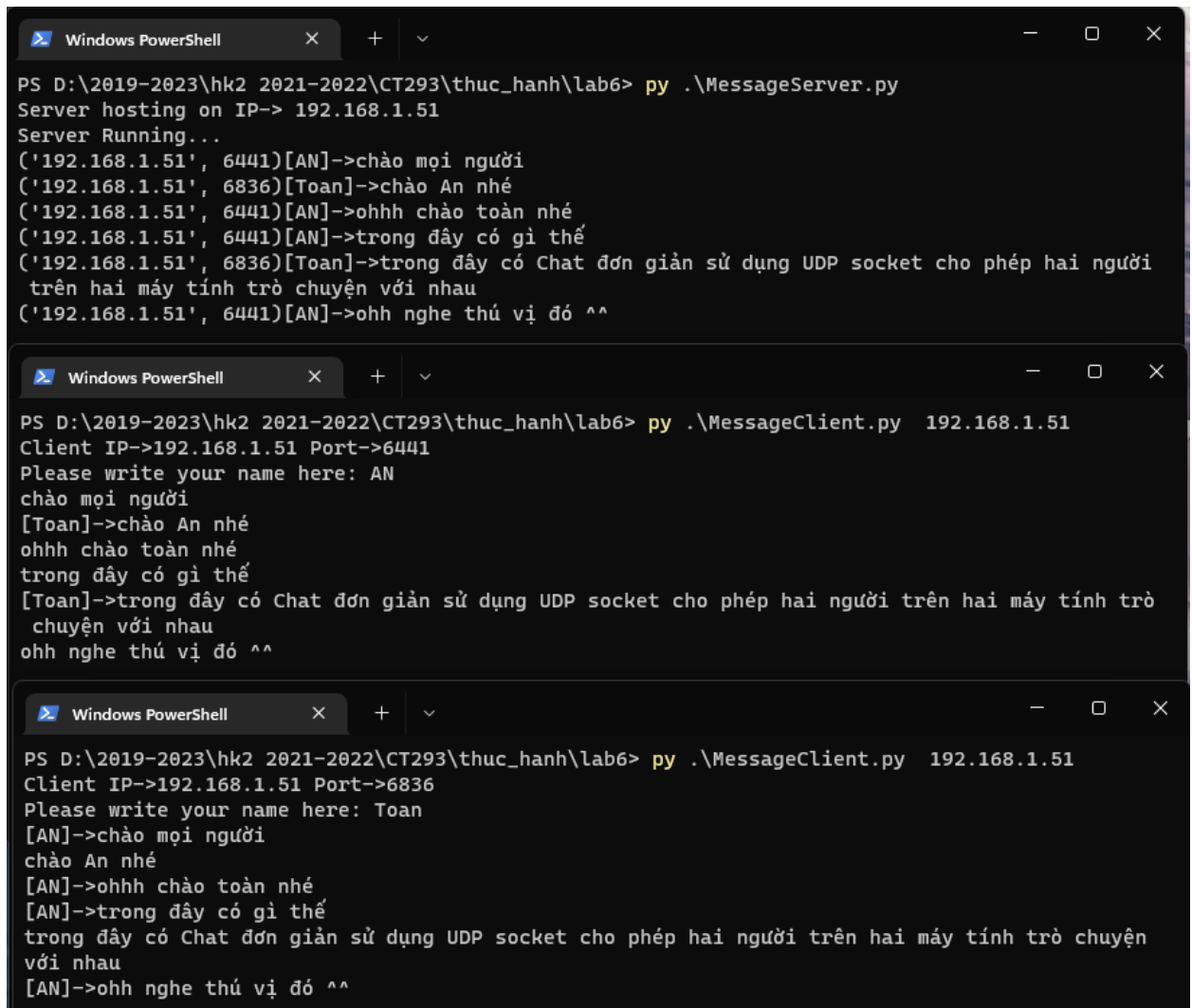
- Code minh họa server:

```
1  import socket
2  import threading
3  import queue
4  #Server Code
5  def RecvData(sock,recvPackets):
6      while True:
7          data,addr = sock.recvfrom(1024)
8          recvPackets.put((data,addr))
9
10     host = socket.gethostbyname(socket.gethostname())
11     port = 5000
12     print('Server hosting on IP-> '+str(host))
13     s = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
14     s.bind((host,port))
15     clients = set()
16     recvPackets = queue.Queue()
17
18     print('Server Running...')
19
20     threading.Thread(target=RecvData,args=(s,recvPackets)).start()
21
22     while True:
23         while not recvPackets.empty():
24             data,addr = recvPackets.get()
25             if addr not in clients:
26                 clients.add(addr)
27                 continue
28             clients.add(addr)
29             data = data.decode('utf-8')
30             if data.endswith('qqq'):
31                 clients.remove(addr)
32                 continue
33             print(str(addr)+data)
34             for c in clients:
35                 if c!=addr:
36                     s.sendto(data.encode('utf-8'),c)
37     s.close()
38     #Serevr Code Ends Here
```


- Code minh họa bên client

```
1  import socket
2  import threading
3  import sys
4  import random
5  import os
6  #Client Code
7  def ReceiveData(sock):
8      while True:
9          try:
10             data,addr = sock.recvfrom(1024)
11             print(data.decode('utf-8'))
12         except:
13             pass
14
15  def RunClient(serverIP):
16      host = socket.gethostbyname(socket.gethostname())
17      port = random.randint(6000,10000)
18      print('Client IP->'+str(host)+' Port->'+str(port))
19      server = (str(serverIP),5000)
20      s = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
21      s.bind((host,port))
22
23      name = input('Please write your name here: ')
24      if name == '':
25          name = 'Guest'+str(random.randint(1000,9999))
26          print('Your name is:'+name)
27      s.sendto(name.encode('utf-8'),server)
28      threading.Thread(target=ReceiveData,args=(s,)).start()
29      while True:
30          data = input()
31          if data == 'qqq':
32              break
33          elif data=='':
34              continue
35          data = '['+name+']' + '->' + data
36          s.sendto(data.encode('utf-8'),server)
37      s.sendto(data.encode('utf-8'),server)
38      s.close()
39      os._exit(1)
```

- Kết quả chạy thử:



```
PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py .\MessageServer.py
Server hosting on IP-> 192.168.1.51
Server Running...
('192.168.1.51', 6441)[AN]->chào mọi người
('192.168.1.51', 6836)[Toan]->chào An nhé
('192.168.1.51', 6441)[AN]->ohhh chào toàn nhé
('192.168.1.51', 6441)[AN]->trong đây có gì thế
('192.168.1.51', 6836)[Toan]->trong đây có Chat đơn giản sử dụng UDP socket cho phép hai người
trên hai máy tính trò chuyện với nhau
('192.168.1.51', 6441)[AN]->ohh nghe thú vị đó ^^

PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py .\MessageClient.py 192.168.1.51
Client IP->192.168.1.51 Port->6441
Please write your name here: AN
chào mọi người
[Toan]->chào An nhé
ohhh chào toàn nhé
trong đây có gì thế
[Toan]->trong đây có Chat đơn giản sử dụng UDP socket cho phép hai người trên hai máy tính trò
chuyện với nhau
ohh nghe thú vị đó ^^

PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py .\MessageClient.py 192.168.1.51
Client IP->192.168.1.51 Port->6836
Please write your name here: Toan
[AN]->chào mọi người
chào An nhé
[AN]->ohhh chào toàn nhé
[AN]->trong đây có gì thế
trong đây có Chat đơn giản sử dụng UDP socket cho phép hai người trên hai máy tính trò chuyện
với nhau
[AN]->ohh nghe thú vị đó ^^
```

3. Bài 3

Sử dụng các thư viện hỗ trợ lập trình mạng ở mức độ High-level của Python để viết một chương trình EmailCrawler cho phép tìm các địa chỉ email trên một website.

- Code bài làm

```
1  '''
2  web crawler for extracting email addresses from web pages.
3
4  Takes a string of URLs and requests each page, checks to see if we've
5  found any emails and prints each email it finds.
6  '''
7
8  import argparse
9  import re
10 import sys
11 import urllib.request
12
13 class Crawler(object):
14
15     def __init__(self, urls):
16         '''
17         @urls: a string containing the (comma separated) URLs to crawl.
18         '''
19         self.urls = urls.split(',')
20
21     def crawl(self):
22         '''
23         Iterate the list of URLs and request each page, then parse it and
24         print the emails we find.
25         '''
26         for url in self.urls:
27             data = self.request(url)
28             #print(data)
29             for email in self.process(data):
30                 print(email)
31
32     @staticmethod
33     def request(url):
34         '''
35         Request @url and return the page contents.
36         '''
37         response = urllib.request.urlopen(url)
38         return response.read().decode("utf-8", "ignore")
39
```

```
40     @staticmethod
41     def process(data):
42         '''
43         Process @data and yield the emails we find in it.
44         '''
45         for email in re.findall(r'[\w.+-]+@[ \w-]+\.[\w.-]+', data):
46             yield email
47
48
49 def main():
50     argparser = argparse.ArgumentParser()
51     argparser.add_argument(
52         '--urls', dest='urls', required=True,
53         help='A comma separated string of emails.')
54     parsed_args = argparser.parse_args()
55     crawler = Crawler(parsed_args.urls)
56     crawler.crawl()
57
58 if __name__ == '__main__':
59     sys.exit(main())
```

- Kết quả thực thi:

```
PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py EmailCrawler.py --urls https://phuclong.com.vn/
Sales@phuclong.com.vn
Info@phuclong.com.vn
PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6> py EmailCrawler.py --urls https://dkmh3.ctu.edu.vn/login.php
dhct@ctu.edu.vn.
PS D:\2019-2023\hk2 2021-2022\CT293\thuc_hanh\lab6>
```

(Chụp hình minh họa code bài làm và kết quả thực thi; đính kèm tập tin code khi nộp bài.)

--- Hết ---