

LAB 6

Concurrent Server - High-level Network programming



Họ tên và MSSV: Lê Nguyễn Phi Nhung B1910426

Nhóm học phần: CT293_01

- *Các sinh viên bị phát hiện sao chép bài của nhau sẽ nhận 0đ cho tất cả bài thực hành của môn này.*
- *Bài nộp phải ở dạng PDF, hình minh họa phải rõ ràng chi tiết. Hình minh họa chỉ cần chụp ở nội dung thực hiện, không chụp toàn màn hình.*

1. Bài 1

Cập nhật phần Server của Bài 2-Lab05 và Bài 4-Lab05 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.)

- Server của BaiTap2-LAB5:

```
SERVER.py X
BaiTap1 > BaiTap2-LAB5 > SERVER.py > ...
1  import socket
2  from threading import Thread
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
11  def run(self):
12      number = self.sock.recv(1024)
13      numberArray={
14          "0" : "Không",
15          "1" : "Một",
16          "2" : "Hai",
17          "3" : "Ba",
18          "4" : "Bốn",
19          "5" : "Năm",
20          "6" : "Sáu",
21          "7" : "Bảy",
22          "8" : "Tám",
23          "9" : "Chín",
24          "10" : "Mười"
25      }
26      dt = numberArray.get(number.decode(), "Không phải số nguyên !!!")
27      self.sock.send(dt.encode())
28
29
30  host = socket.gethostname()
31  p = 8888
32  tcpsock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
33  tcpsock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
34  tcpsock.bind((host, p))
35  threads = []
36
37  while True:
38      tcpsock.listen(5)
39      print('Server is listening...')
40      (conn, (ip, port)) = tcpsock.accept()
41      print('Got connection from ', (ip, port))
42      newthread = ClientThread(ip, port, conn)
43      newthread.start()
44      threads.append(newthread)
```

- Kết quả:

```
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6> & "C:/Program Files/Python36/python.exe"
"c:/Users/PC Nhung/Documents/B1910426_CODELAB6/BaiTap1/BaiTap2-LAB5/SERVER.py"
Server is listening...
Got connection from ('192.168.2.108', 55869)
New thread started for 192.168.2.108:55869
Server is listening...
Got connection from ('192.168.2.108', 55872)
New thread started for 192.168.2.108:55872
Server is listening...
```

```
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6> cd BaiTap1\BaiTap2-LAB5
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap2-LAB5> python CLIENT.py
Nhập vào số nguyên: 7
Bảy
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap2-LAB5> python CLIENT.py
Nhập vào số nguyên: 20
Không phải số nguyên !!!
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap2-LAB5> []
```

- Server của BaiTap4-LAB5:

```
SERVER.py X
BaiTap1 > BaiTap4-LAB5 > SERVER.py > ...
1  from ast import Break
2  from email import message
3  import os
4  import socket
5  from threading import Thread
6
7  class ClientThread(Thread):
8      def __init__(self, ip, port, sock1, sock2):
9          Thread.__init__(self)
10         self.ip = ip
11         self.port = port
12         self.sock1 = sock1
13         self.sock2 = sock2
14         print(" New thread started for "+ip+": "+str(port))
15
16     def run(self):
17         SuccessMess="OK\n"
18         ErrorMess="ERROR\n"
19         message = self.sock1.recv(1024).decode()
20         command, filename = message.split(" ")
21         if(os.path.isfile(filename) | os.path.isdir(filename)):
22             self.sock1.send(SuccessMess.encode())
23             if(command == "GET"):
24                 fi = open(filename,'rb')
25                 while True:
26                     data = fi.read(buffer_size)
27                     while (data):
28                         self.sock2.send(data)
29                         data = fi.read(buffer_size)
30                     if not data:
31                         fi.close()
32                         self.sock2.close()
33                         break
34             elif(command == "DELETE"):
35                 os.remove(filename)
36             elif(command == "LIST"):
37                 arrs=os.listdir(filename)
38                 list = ''
39                 for arr in arrs:
40                     list += arr+' '
41                 self.sock2.send(list.encode())
42                 self.sock2.close()
43             else:
44                 self.sock1.send(ErrorMess.encode())
45
46 s1 = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
47 s1.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
48 s2 = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
49 s2.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
50
51 host = socket.gethostname()
52 port1 = 8000
53 port2 = 8001
54 buffer_size = 1024
55 s1.bind((host, port1))
56 s2.bind((host, port2))
57
58 while True:
59     s1.listen(5)
60     s2.listen(5)
61     print('Server is listening...')
62     (conn1, (ip1, port1)) = s1.accept()
63     (conn2, (ip2, port2)) = s2.accept()
64     print('Got connection from', (ip1, port1))
65     newthread = ClientThread(ip1, port1, conn1, conn2)
66     newthread.start()
```

- Kết quả:

```
LeNhung.txt X
BaiTap1 > BaiTap4-LAB5 > LeNhung.txt
1  Họ tên: Lê Nguyễn Phi Nhung
2  MSSV: B1910426
```

```
Windows PowerShell
Copyright (c) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6> cd BaiTap1\BaiTap4-LAB5
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap4-LAB5> python SERVER.py
Server is listening...
Got connection from ('192.168.56.1', 55199)
New thread started for 192.168.56.1:55199
Server is listening...
Got connection from ('192.168.56.1', 55270)
New thread started for 192.168.56.1:55270
Server is listening...
Got connection from ('192.168.56.1', 55329)
New thread started for 192.168.56.1:55329
Server is listening...
[]

PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap4-LAB5> python CLIENT.py
Vui lòng nhập câu lệnh: GET LeNhung.txt
OK

Họ tên: Lê Nguyễn Phi Nhung
MSSV: B1910426

PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap4-LAB5> python CLIENT.py
Vui lòng nhập câu lệnh: DELETE LeNhung.txt
OK

PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap4-LAB5> python CLIENT.py
Vui lòng nhập câu lệnh: LIST Nhung
OK

B1910426.txt
Lê
Nguyễn
Phi
Nhung.txt

PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap1\BaiTap4-LAB5>
```

```
teNhung.txt X
BaiTap1 > BaiTap4-LAB5 > teNhung.txt
1  Họ tên: Lê Nguyễn Phi Nhung
2  MSSV: B1910426|
```

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.)

- Client1:

```
CLIENT1.py X
BaiTap2 > CLIENT1.py > ...
1  from threading import Thread
2  import socket
3
4  host = socket.gethostname()
5  port = 8001
6
7  class RecvThread(Thread):
8      def __init__(self, host, port, sock):
9          Thread.__init__(self)
10         self.host = host
11         self.port = port
12         self.sock = sock
13
14     def run(self):
15         while True:
16             data, addr = self.sock.recvfrom(1024)
17             data = data.decode()
18             print(f"[Đã nhận] {data}\n")
19
20     class SendThread(Thread):
21         def __init__(self, host, port, sock, data):
22             Thread.__init__(self)
23             self.host = host
24             self.port = port
25             self.sock = sock
26             self.data = data
27
28         def run(self):
29             self.sock.sendto(self.data.encode(), (self.host, self.port))
30
31     udp = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
32     udp.bind((host, port))
33
34     r = RecvThread(host, port, udp)
35     r.start()
36
37     print("\tBẮT ĐẦU ĐOẠN CHAT")
38     while True:
39         data = input()
40         if not data:
41             break
42         s = SendThread(host, 8000, udp, data)
43         s.start()
```

- Client2:

```
CLIENT2.py X
BaiTap2 > CLIENT2.py > ...
1  from threading import Thread
2  import socket
3
4  host = socket.gethostname()
5  port = 8000
6
7  class RecvThread(Thread):
8      def __init__(self, host, port, sock):
9          Thread.__init__(self)
10         self.host = host
11         self.port = port
12         self.sock = sock
13
14     def run(self):
15         while True:
16             data, addr = self.sock.recvfrom(1024)
17             data = data.decode()
18             print(f"[Đã nhận] {data}\n")
19
20 class SendThread(Thread):
21     def __init__(self, host, port, sock, data):
22         Thread.__init__(self)
23         self.host = host
24         self.port = port
25         self.sock = sock
26         self.data = data
27
28     def run(self):
29         self.sock.sendto(self.data.encode(), (self.host, self.port))
30
31 udp = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
32 udp.bind((host, port))
33
34 r = RecvThread(host, port, udp)
35 r.start()
36
37 print("\tBẮT ĐẦU ĐOẠN CHAT")
38 while True:
39
40     data = input()
41     if not data:
42         break
43     s = SendThread(host, 8001, udp, data)
44     s.start()
```

- Kết quả khi thực hiện đoạn chat:

```
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6> cd BaiTap2
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap2> python CLIENT1.py
BẮT ĐẦU ĐOẠN CHAT

Xin Chào CLIENT2

Mình rất vui khi được làm quen với bạn

[Đã nhận] Chào CLIENT1

[Đã nhận] Mình cũng vui khi trò chuyện với bạn

Chúng mình sẽ gặp lại vào lần sau nhé

[Đã nhận] Được, lần sau lại trò chuyện tiếp

[Đã nhận] Goodbye
```

```
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6> cd BaiTap2
PS C:\Users\PC Nhung\Documents\B1910426_CODELAB6\BaiTap2> python CLIENT2.py
    BẮT ĐẦU ĐOẠN CHAT
[Đã nhận] Xin Chào CLIENT2

[Đã nhận] Mình rất vui khi được làm quen với bạn

Chào CLIENT1
Mình cũng vui khi trò chuyện với bạn
[Đã nhận] Chúng mình sẽ gặp lại vào lần sau nhé

Được, lần sau lại trò chuyện tiếp
Goodbye
```

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.

Gợi ý:

- (1) Sử dụng hàm `urllib.request.urlopen` để load trang chủ của 1 website.
 - (2) Tìm trong nội dung trang web tải xuống các email và URL. Đưa email và URL vào danh sách.
 - (3) Tiếp tục tải các trang web có trong website sử dụng các URL tìm được trong danh sách. Lập lại đến khi tất cả hết các URL.
 - Có thể sử dụng hàm `re.findall` để tìm các email và URL các trang web. Ví dụ:
- ```
+ emails = re.findall(r'[\w.+~]+@[\w-]+\.[\w.-]+', text)
+ urls = re.findall(
 ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[*\(\)\,]|\
 (?:%[0-9a-fA-F][0-9a-fA-F]))+', text)
```

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

```

1 from urllib import request
2 import re
3
4 def get_emails_urls(url):
5 r = request.urlopen(url)
6 text = r.read().decode(r.headers.get_content_charset() if r.headers.get_content_charset() else 'utf-8')
7 return re.findall(r'[\w.+-]+@[\w-]+\.[\w.-]+' , text), \
8 re.findall('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[*\(\)\,])\
9 (?:%[0-9a-fA-F][0-9a-fA-F])+', text)
10
11 url = input("Vui lòng nhập URL: ")
12
13 emails, urls = get_emails_urls(url)
14
15 while urls:
16 try:|
17 url = urls.pop(0)
18 e, u = get_emails_urls(url)
19 emails.extend(e)
20 except Exception as e:
21 e
22
23 print(emails)

```

- Kết quả

[illegible]

--- Hết ---