REG NO : 3122235002128

NAME : SOWKANDHA RAGHAV G

DEPT : IT – C

NETWORK PROGRAMMING LAB

EX NO : 3

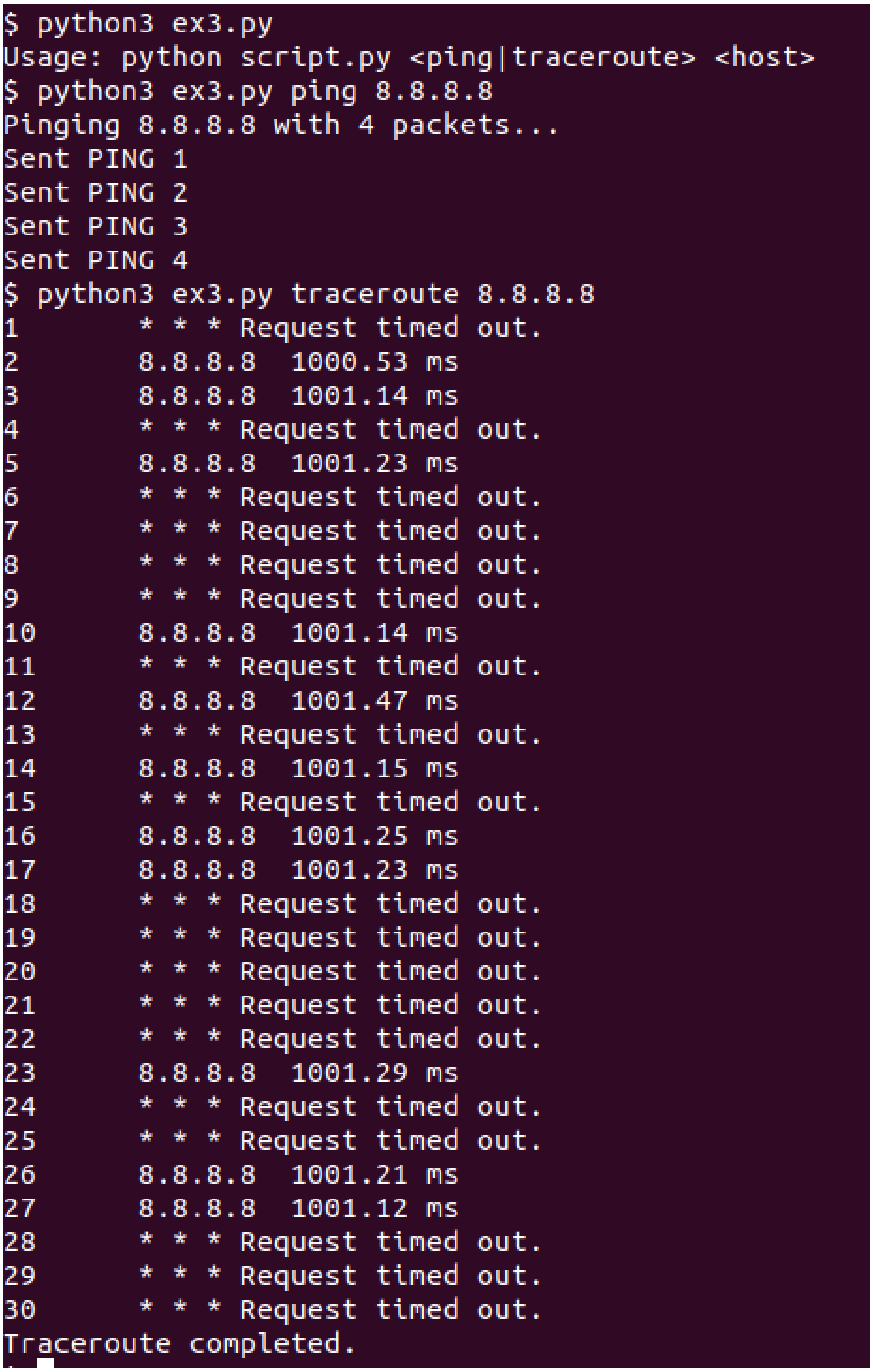
PING AND TRACEROUTE

Write a code simulating PING and TRACEROUTE commands using Twisted Python.

CODING:

```
from twisted.internet import reactor,defer
from twisted.internet.protocol import DatagramProtocol
import random
import time
import sys
class PingClient(DatagramProtocol):
   def __init__(self,host,count=4):
      self.host=host
      self.count=count
      self.seq=0
      self.delays=[]
   def startProtocol(self):
      print(f"Pinging {self.host} with {self.count} packets...")
      self.transport.connect(self.host,33434)
      self.sendPing()
   def sendPing(self):
      if self.seq<self.count:
         self.seq+=1
         self.transport.write(b"PING")
         print(f"Sent PING {self.seq}")
         self.sent_time=time.time()
         reactor.callLater(1,self.sendPing)
      else:
         reactor.callLater(2,self.stop)
   def datagramReceived(self,data,addr):
      rtt=(time.time()-self.sent_time)*1000
      self.delays.append(rtt)
      print(f"Reply from {addr[0]}: seq={self.seq} time={rtt:.2f} ms")
   def stop(self):
      if self.delays:
            print(f"Ping statistics for {self.host}: min={min(self.delays):.2f} ms, "
                  f"max={max(self.delays):.2f} ms, avg={sum(self.delays)/len(self.delays):.2f} ms")
        reactor.stop()
class TracerouteClient(DatagramProtocol):
    def __init__(self,host,max_hops=30):
        self.host=host
        self.ttl=1
        self.max_hops=max_hops
    def startProtocol(self):
        self.transport.connect(self.host,33434)
        self.sendPacket()
    def sendPacket(self):
        if self.ttl>self.max_hops:
            print("Traceroute completed.")
            reactor.stop()
            return
        self.transport.write(b"TRACE")
        start_time=time.time()
        reactor.callLater(1,self.checkResponse,start_time)
    def checkResponse(self,start_time):
        rtt=(time.time()-start_time)*1000
        if random.random() > 0.3:
            print(f"{self.ttl}\t* * * Request timed out.")
        else:
            print(f"{self.ttl}\t{self.host}  {rtt:.2f} ms")
        self.ttl+=1
        self.sendPacket()
if __name__=='__main__':
    if len(sys.argv) < 3:
        print("Usage: python script.py <ping|traceroute> <host>")
        sys.exit(1)
    command,host=sys.argv[1],sys.argv[2]
    if command.lower()=="ping":
        reactor.listenUDP(0,PingClient(host))
    elif command.lower()=="traceroute":
        reactor.listenUDP(0,TracerouteClient(host))
    else:
        print("Invalid command. Use 'ping' or 'traceroute'.")
        sys.exit(1)
    reactor.run()
```

OUTPUT:



```
$ python3 ex3.py
Usage: python script.py <ping|traceroute> <host>
$ python3 ex3.py ping 8.8.8.8
Pinging 8.8.8.8 with 4 packets...
Sent PING 1
Sent PING 2
Sent PING 3
Sent PING 4
$ python3 ex3.py traceroute 8.8.8.8
1       * * * Request timed out.
2       8.8.8.8  1000.53 ms
3       8.8.8.8  1001.14 ms
4       * * * Request timed out.
5       8.8.8.8  1001.23 ms
6       * * * Request timed out.
7       * * * Request timed out.
8       * * * Request timed out.
9       * * * Request timed out.
10      8.8.8.8  1001.14 ms
11      * * * Request timed out.
12      8.8.8.8  1001.47 ms
13      * * * Request timed out.
14      8.8.8.8  1001.15 ms
15      * * * Request timed out.
16      8.8.8.8  1001.25 ms
17      8.8.8.8  1001.23 ms
18      * * * Request timed out.
19      * * * Request timed out.
20      * * * Request timed out.
21      * * * Request timed out.
22      * * * Request timed out.
23      8.8.8.8  1001.29 ms
24      * * * Request timed out.
25      * * * Request timed out.
26      8.8.8.8  1001.21 ms
27      8.8.8.8  1001.12 ms
28      * * * Request timed out.
29      * * * Request timed out.
30      * * * Request timed out.
Traceroute completed.
```