Introduction to Web Science

Assignment 1

Prof. Dr. Steffen Staab
staab@uni-koblenz.de

René Pickhardt
rpickhardt@uni-koblenz.de

Korok Sengupta
koroksengupta@uni-koblenz.de

Institute of Web Science and Technologies
Department of Computer Science
University of Koblenz-Landau

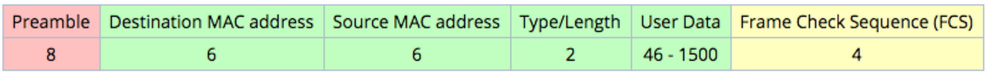
Submission until: November 2, 2016, 10:00 a.m.
Tutorial on: November 4th, 2016, 12:00 p.m.

The main objective of this assignment is for you to use different tools with which you can understand the network that you are connected to or you are connecting to in a better sense. These tasks are not always specific to "Introduction to Web Science". For all the assignment questions that require you to write a code, make sure to include the code in the answer sheet, along with a separate python file. Where screen shots are required, please add them in the answers directly and not as separate files.

Group name: uniform
Group members: Pradip Giri, Jalak Arvind Kumar Pansuriya, Madhu Rakhal Magar

1 Ethernet Frame (5 Points)

Ethernet Frame is of the given structure:



Preamble	Destination MAC address	Source MAC address	Type/Length	User Data	Frame Check Sequence (FCS)
8	6	6	2	46 - 1500	4

Figure 1: Ethernet Frame Structure

Given below is an Ethernet frame without the Preamble and the Frame Check Sequence.

```
00 27 10 21 fa 48 00 13     10 e8 dd 52 08 06 00 01
08 00 06 04 00 01 00 13     10 e8 dd 52 c0 a8 02 01
00 00 00 00 00 00 c0 a8     02 67
```

Find:

1. Source MAC Address
2. Destination MAC Address
3. What protocol is inside the data payload?
4. Please mention what the last 2 fields hold in the above frame.

Answers:

1. 00 13 10 e8 dd 52
2. 00 27 10 21 fa 48
3. 0x0806 Address Resolution Protocol
4. The last two fields in the above frame holds Target Protocol Address(TPA)

2 Cable Issue (5 Points)

Let us consider we have two cables of 20 meters each. One of them is in a 100MBps network while the other is in a 10MBps network. If you had to transfer data through each of them, how much time it would take for the first bit to arrive in each setting? (For your calculation you can assume that the speed of light takes the same value as in the videos.) Please provide formulas and calculatoins along with your results.

Answers:

1. For 100MBps network
   100MBps = 100 Million bits per second
   For 1 bit the time will be
   (1 second/100 Million bits per second) = 0.00000001 seconds = 10 nanoseconds
   To send 1 bit of data it takes 10 nanoseconds .

   The speed of light is 299792458 meters per second = electromagnetic waves (30M m/s approx)
   So, in 100MBit cable, 1 bit will travel
   (3M meters / 1 second)*(10 nanoseconds) = 3 meters in one clock cycle .
   So, to travel 20 meters it will take
   (20*10)/3 = 66.67 nanoseconds

2. For 10MBps network
   10MBps = 10 Million bits per seconds
   For 1 bit the time will be
   (1second/10 Million bits per second) = 0.0000001 seconds = 100 nanoseconds
   To send 1 bit of data it takes 100 nanoseconds in this network.

   We consider speed of light is same as above (i.e 30M m/s approx) So, in 10 MBps network cable, 1 bit will travel
   (3M meters / 1 second)*(100 nanoseconds) = 30 meters in one clock cycle.
   So, to travel 20 meters it will take
   (20*100)/30 = 66.67 nanoseconds.

   Basically, it takes equal time for both cases.

3 Basic Network Tools (10 Points)

Listed below are some of the commands which you need to ”google” to understand what they stand for:

1. *ipconfig / ifconfig*
2. *ping*
3. *traceroute*
4. *arp*
5. *dig*

Consider a situation in which you need to check if `www.wikipedia.org` is reachable or not. Using the knowledge you gained above to find the following information:

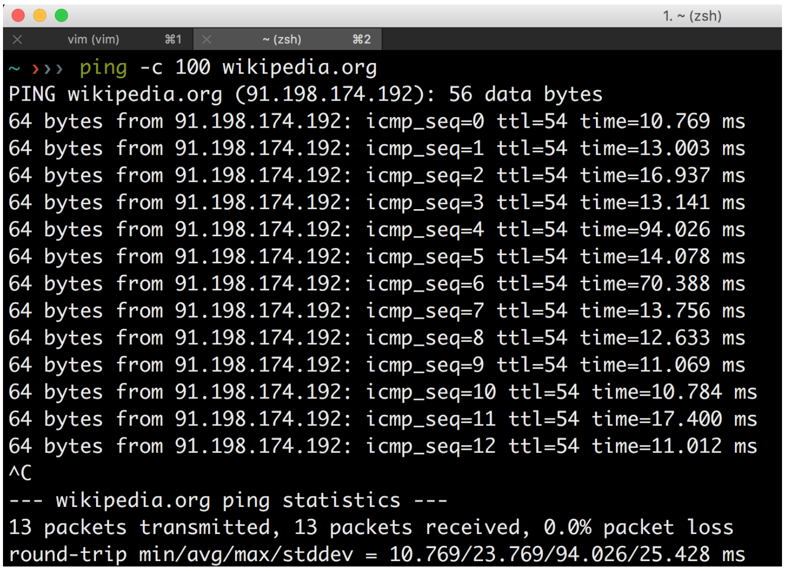
1. The *% packet loss* if at all it happened after sending 100 packets.
2. *Size* of the packet sent to *Wikipedia* server
3. *IP address* of your machine and the *Wikipedia* server
4. *Query Time* for DNS query of the above url.
5. Number of *Hops* in between your machine and the server
6. MAC address of the device that is acting as your network gateway.

Do this once in the university and once in your home/dormitory network. With your answers, you must paste the screen shots to validate your find.

Answers:

1. The *% packet loss* if at all it happened after sending 100 packets.

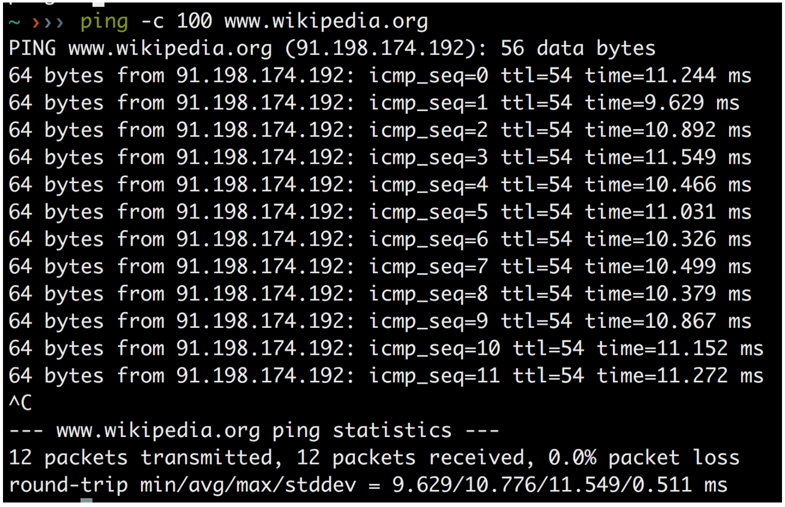
   a) Home network:
   0% Package loss after sending 100 packages (ping -c 100 www.wikipedia.org)



```
~ ››› ping -c 100 wikipedia.org
PING wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=54 time=10.769 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=54 time=13.003 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=54 time=16.937 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=54 time=13.141 ms
64 bytes from 91.198.174.192: icmp_seq=4 ttl=54 time=94.026 ms
64 bytes from 91.198.174.192: icmp_seq=5 ttl=54 time=14.078 ms
64 bytes from 91.198.174.192: icmp_seq=6 ttl=54 time=70.388 ms
64 bytes from 91.198.174.192: icmp_seq=7 ttl=54 time=13.756 ms
64 bytes from 91.198.174.192: icmp_seq=8 ttl=54 time=12.633 ms
64 bytes from 91.198.174.192: icmp_seq=9 ttl=54 time=11.069 ms
64 bytes from 91.198.174.192: icmp_seq=10 ttl=54 time=10.784 ms
64 bytes from 91.198.174.192: icmp_seq=11 ttl=54 time=17.400 ms
64 bytes from 91.198.174.192: icmp_seq=12 ttl=54 time=11.012 ms
^C
--- wikipedia.org ping statistics ---
13 packets transmitted, 13 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 10.769/23.769/94.026/25.428 ms
```

Figure 2: At dorm.

b) University's network:
0% Package loss after sending 100 packages (ping -c 100 www.wikipedia.org)

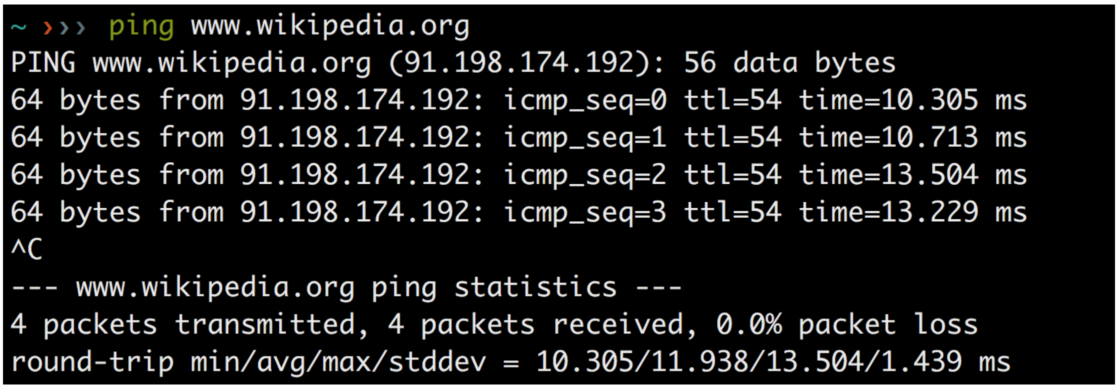


```
~ ››› ping -c 100 www.wikipedia.org
PING www.wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=54 time=11.244 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=54 time=9.629 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=54 time=10.892 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=54 time=11.549 ms
64 bytes from 91.198.174.192: icmp_seq=4 ttl=54 time=10.466 ms
64 bytes from 91.198.174.192: icmp_seq=5 ttl=54 time=11.031 ms
64 bytes from 91.198.174.192: icmp_seq=6 ttl=54 time=10.326 ms
64 bytes from 91.198.174.192: icmp_seq=7 ttl=54 time=10.499 ms
64 bytes from 91.198.174.192: icmp_seq=8 ttl=54 time=10.379 ms
64 bytes from 91.198.174.192: icmp_seq=9 ttl=54 time=10.867 ms
64 bytes from 91.198.174.192: icmp_seq=10 ttl=54 time=11.152 ms
64 bytes from 91.198.174.192: icmp_seq=11 ttl=54 time=11.272 ms
^C
--- www.wikipedia.org ping statistics ---
12 packets transmitted, 12 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 9.629/10.776/11.549/0.511 ms
```

Figure 3: At University.

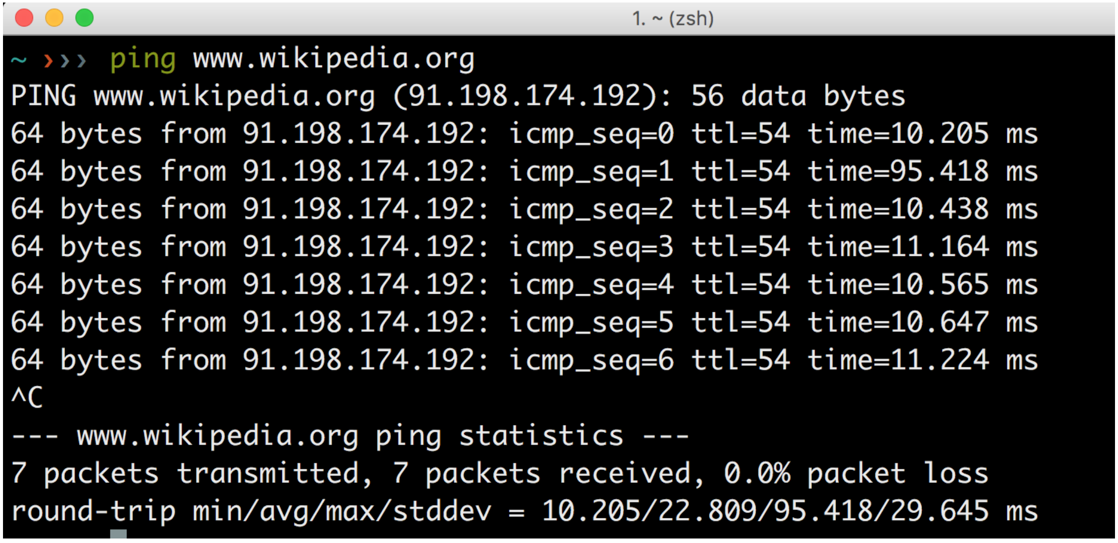
2. Size of the packet sent to Wikipedia server

   a) Home network:
   Size of the package sent to www.wikipedia.org is 56bytes (ping www.wikipedia.org)



```
~ ››› ping www.wikipedia.org
PING www.wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=54 time=10.305 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=54 time=10.713 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=54 time=13.504 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=54 time=13.229 ms
^C
--- www.wikipedia.org ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 10.305/11.938/13.504/1.439 ms
```

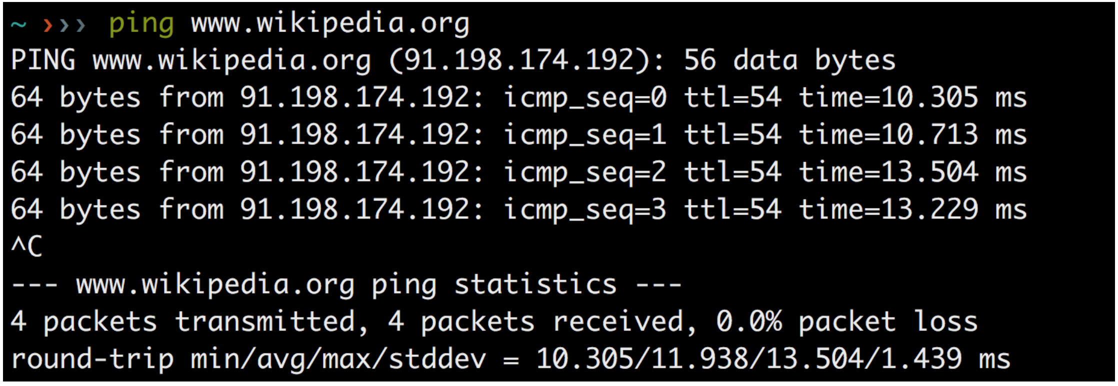
b) University's network: Size of the package sent to www.wikipedia.org is 56bytes (ping www.wikipedia.org)



```
~ ››› ping www.wikipedia.org
PING www.wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=54 time=10.205 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=54 time=95.418 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=54 time=10.438 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=54 time=11.164 ms
64 bytes from 91.198.174.192: icmp_seq=4 ttl=54 time=10.565 ms
64 bytes from 91.198.174.192: icmp_seq=5 ttl=54 time=10.647 ms
64 bytes from 91.198.174.192: icmp_seq=6 ttl=54 time=11.224 ms
^C
--- www.wikipedia.org ping statistics ---
7 packets transmitted, 7 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 10.205/22.809/95.418/29.645 ms
```

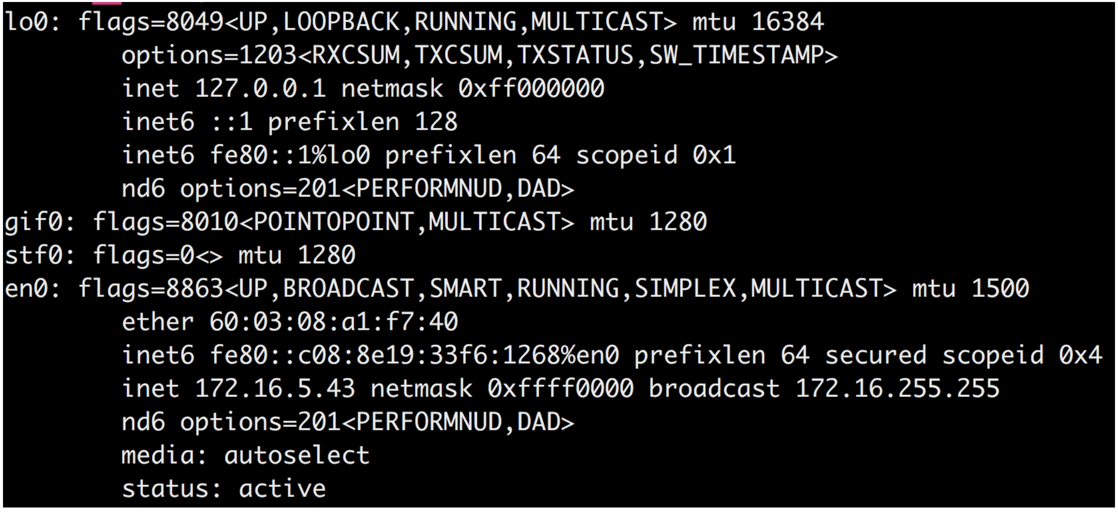
3. IP address of your machine and the Wikipedia server

a) Home network: (ping www.wikipedia.org)
The IP address of wikipedia.org server is [91.198.174.192]



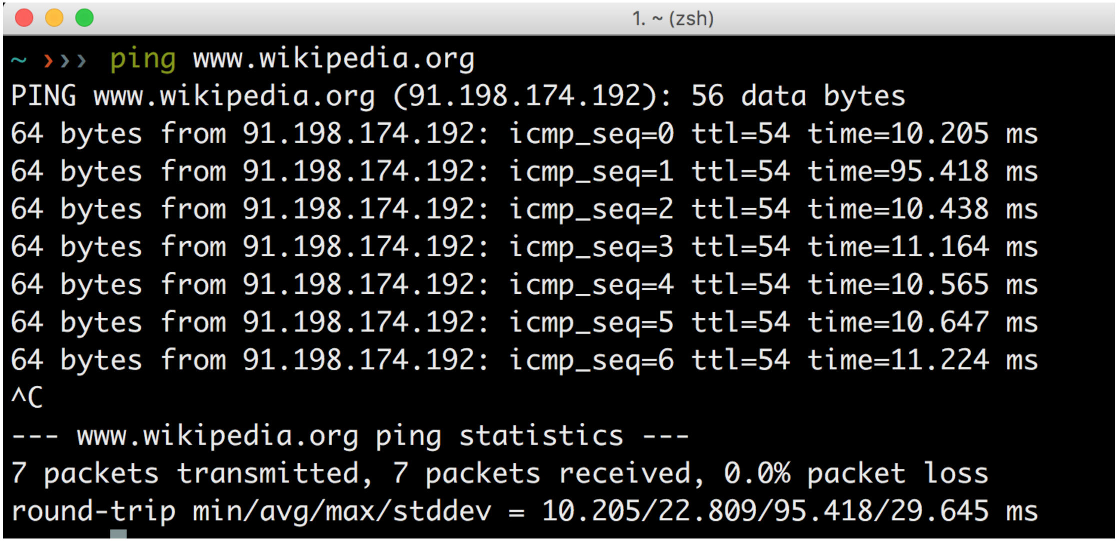
```
~ ››› ping www.wikipedia.org
PING www.wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=54 time=10.305 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=54 time=10.713 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=54 time=13.504 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=54 time=13.229 ms
^C
--- www.wikipedia.org ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 10.305/11.938/13.504/1.439 ms
```

The IP address of local machine is 172.16.5.43



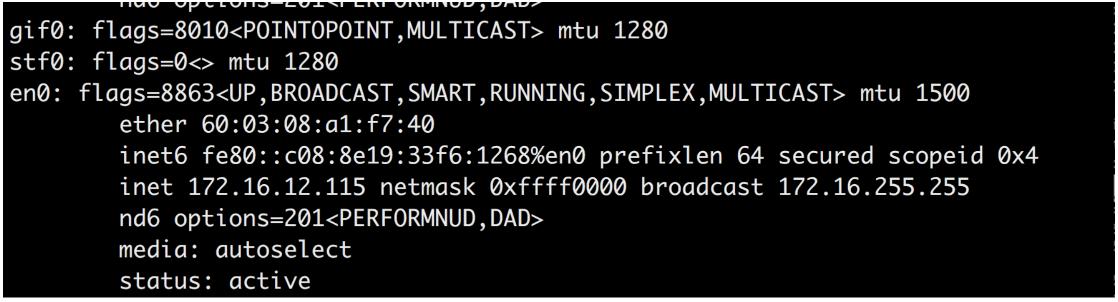
```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 60:03:08:a1:f7:40
        inet6 fe80::c08:8e19:33f6:1268%en0 prefixlen 64 secured scopeid 0x4
        inet 172.16.5.43 netmask 0xffff0000 broadcast 172.16.255.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
```

b) University network: (ping www.wikipedia.org)
The IP address of wikipedia.org server is [91.198.174.192]



```
~ ››› ping www.wikipedia.org
PING www.wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=54 time=10.205 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=54 time=95.418 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=54 time=10.438 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=54 time=11.164 ms
64 bytes from 91.198.174.192: icmp_seq=4 ttl=54 time=10.565 ms
64 bytes from 91.198.174.192: icmp_seq=5 ttl=54 time=10.647 ms
64 bytes from 91.198.174.192: icmp_seq=6 ttl=54 time=11.224 ms
^C
--- www.wikipedia.org ping statistics ---
7 packets transmitted, 7 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 10.205/22.809/95.418/29.645 ms
```

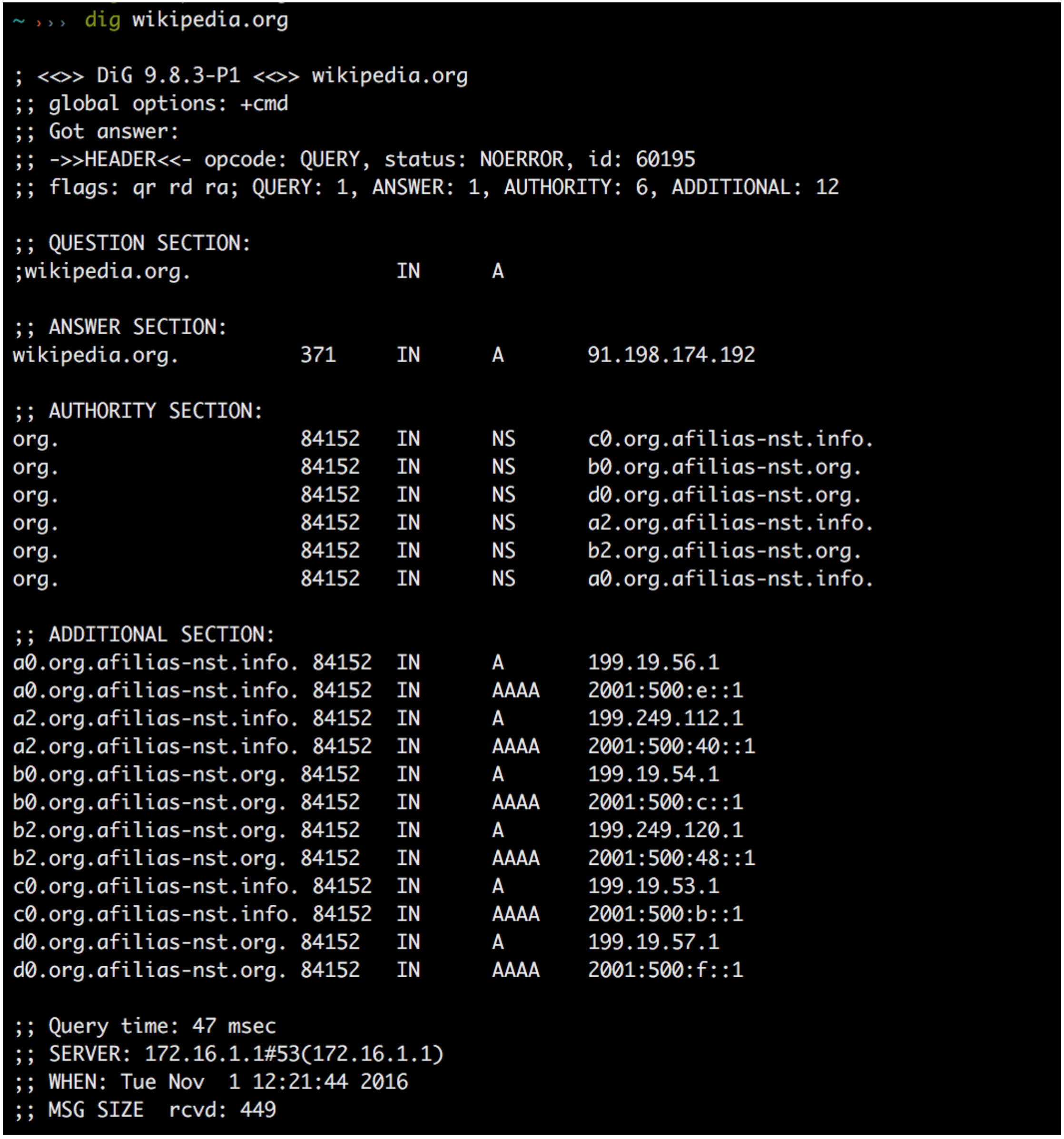
The IP address of local machine is 172.16.12.115



```
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 60:03:08:a1:f7:40
        inet6 fe80::c08:8e19:33f6:1268%en0 prefixlen 64 secured scopeid 0x4
        inet 172.16.12.115 netmask 0xffff0000 broadcast 172.16.255.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
```

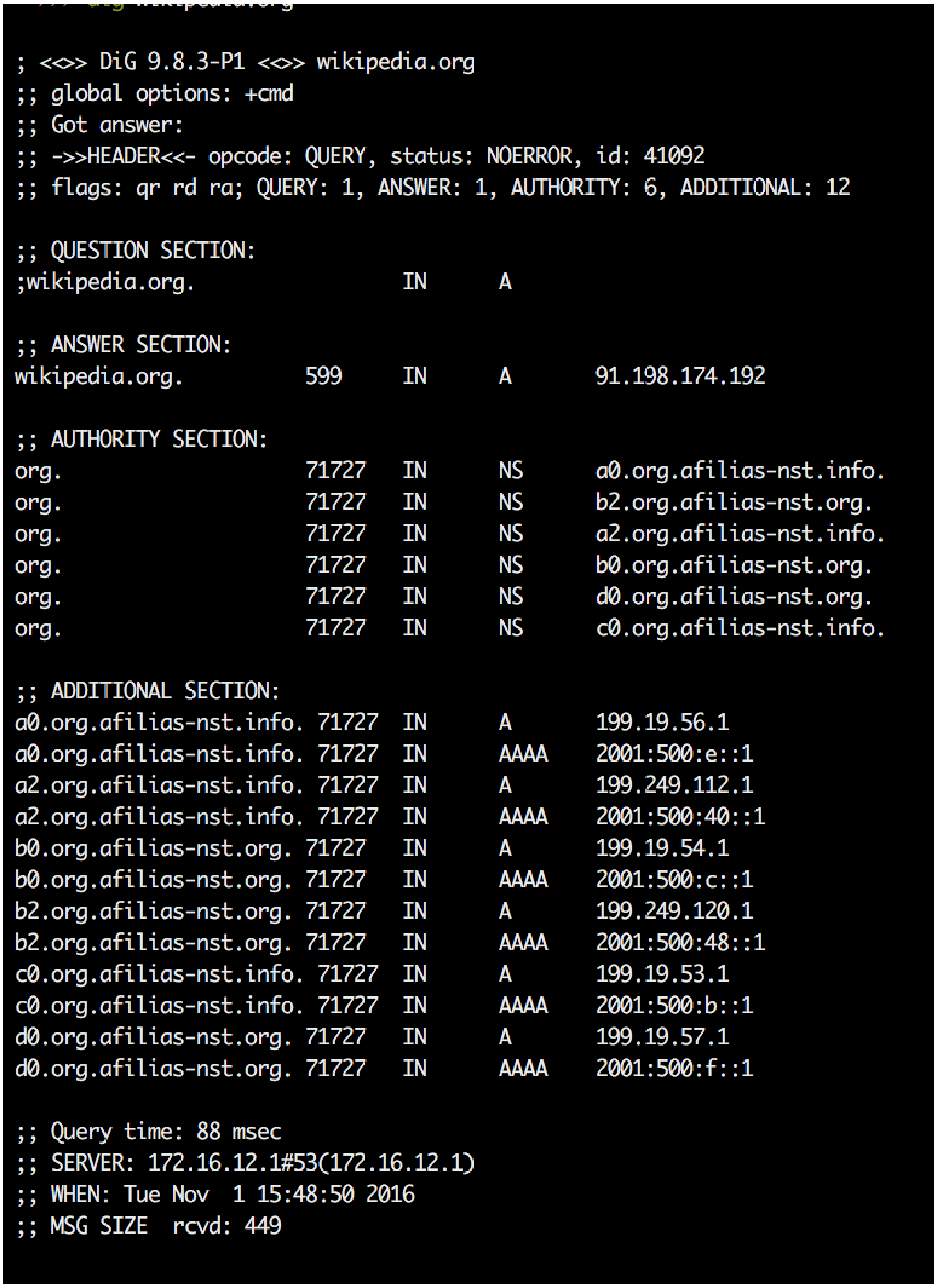
4. Query Time for DNS query of the above url.

a) Home network:
Query time for DNS query of www.wikipedia.org: 47msec (dig www.wikipedia.org)



```
~ ››› dig wikipedia.org

; <<>> DiG 9.8.3-P1 <<>> wikipedia.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60195
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 6, ADDITIONAL: 12

;; QUESTION SECTION:
;wikipedia.org.                 IN      A

;; ANSWER SECTION:
wikipedia.org.          371     IN      A       91.198.174.192

;; AUTHORITY SECTION:
org.                    84152   IN      NS      c0.org.afilias-nst.info.
org.                    84152   IN      NS      b0.org.afilias-nst.org.
org.                    84152   IN      NS      d0.org.afilias-nst.org.
org.                    84152   IN      NS      a2.org.afilias-nst.info.
org.                    84152   IN      NS      b2.org.afilias-nst.org.
org.                    84152   IN      NS      a0.org.afilias-nst.info.

;; ADDITIONAL SECTION:
a0.org.afilias-nst.info. 84152  IN      A       199.19.56.1
a0.org.afilias-nst.info. 84152  IN      AAAA    2001:500:e::1
a2.org.afilias-nst.info. 84152  IN      A       199.249.112.1
a2.org.afilias-nst.info. 84152  IN      AAAA    2001:500:40::1
b0.org.afilias-nst.org. 84152   IN      A       199.19.54.1
b0.org.afilias-nst.org. 84152   IN      AAAA    2001:500:c::1
b2.org.afilias-nst.org. 84152   IN      A       199.249.120.1
b2.org.afilias-nst.org. 84152   IN      AAAA    2001:500:48::1
c0.org.afilias-nst.info. 84152  IN      A       199.19.53.1
c0.org.afilias-nst.info. 84152  IN      AAAA    2001:500:b::1
d0.org.afilias-nst.org. 84152   IN      A       199.19.57.1
d0.org.afilias-nst.org. 84152   IN      AAAA    2001:500:f::1

;; Query time: 47 msec
;; SERVER: 172.16.1.1#53(172.16.1.1)
;; WHEN: Tue Nov  1 12:21:44 2016
;; MSG SIZE  rcvd: 449
```

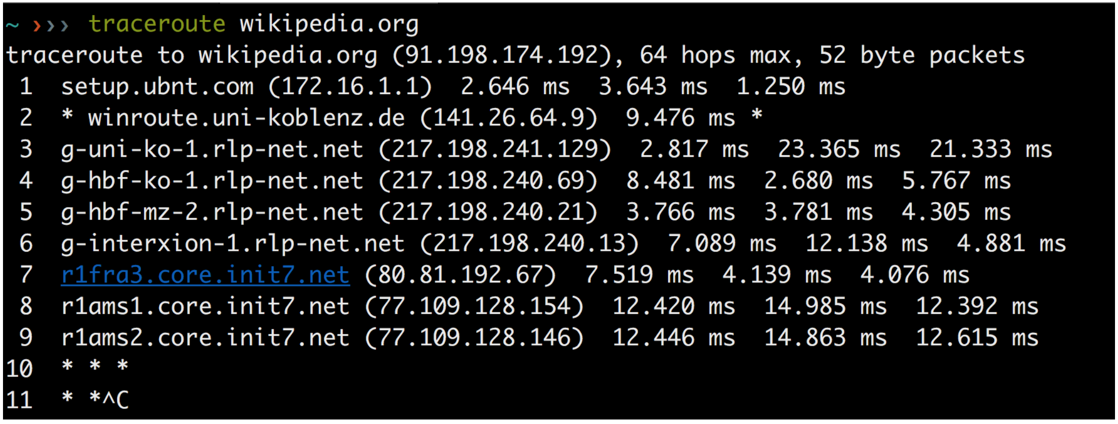
b) University network:
Query time for DNS query of www.wikipedia.org: 88msec (dig www.wikipedia.org)



```
dig wikipedia.org

; <<>> DiG 9.8.3-P1 <<>> wikipedia.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 41092
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 6, ADDITIONAL: 12

;; QUESTION SECTION:
;wikipedia.org.                 IN      A

;; ANSWER SECTION:
wikipedia.org.          599     IN      A       91.198.174.192

;; AUTHORITY SECTION:
org.                    71727   IN      NS      a0.org.afilias-nst.info.
org.                    71727   IN      NS      b2.org.afilias-nst.org.
org.                    71727   IN      NS      a2.org.afilias-nst.info.
org.                    71727   IN      NS      b0.org.afilias-nst.org.
org.                    71727   IN      NS      d0.org.afilias-nst.org.
org.                    71727   IN      NS      c0.org.afilias-nst.info.

;; ADDITIONAL SECTION:
a0.org.afilias-nst.info. 71727  IN      A       199.19.56.1
a0.org.afilias-nst.info. 71727  IN      AAAA    2001:500:e::1
a2.org.afilias-nst.info. 71727  IN      A       199.249.112.1
a2.org.afilias-nst.info. 71727  IN      AAAA    2001:500:40::1
b0.org.afilias-nst.org. 71727   IN      A       199.19.54.1
b0.org.afilias-nst.org. 71727   IN      AAAA    2001:500:c::1
b2.org.afilias-nst.org. 71727   IN      A       199.249.120.1
b2.org.afilias-nst.org. 71727   IN      AAAA    2001:500:48::1
c0.org.afilias-nst.info. 71727  IN      A       199.19.53.1
c0.org.afilias-nst.info. 71727  IN      AAAA    2001:500:b::1
d0.org.afilias-nst.org. 71727   IN      A       199.19.57.1
d0.org.afilias-nst.org. 71727   IN      AAAA    2001:500:f::1

;; Query time: 88 msec
;; SERVER: 172.16.12.1#53(172.16.12.1)
;; WHEN: Tue Nov  1 15:48:50 2016
;; MSG SIZE  rcvd: 449
```

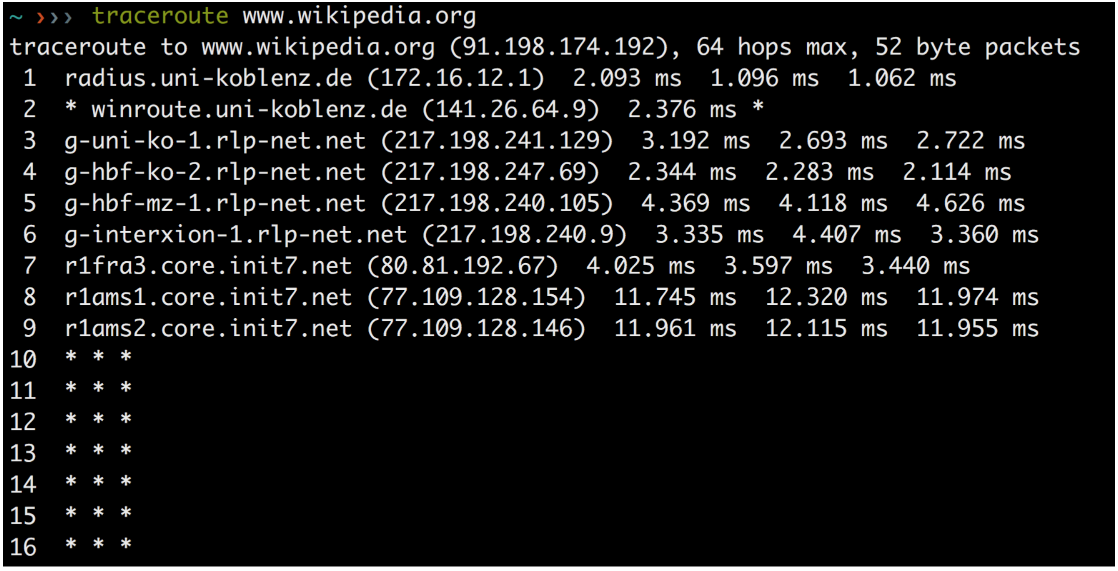
5. Number of *Hops* in between your machine and the server.

a) Home network:
It took 9 Hops to get to the www.wikipedia.org server (traceroute www.wikipedia.org)



```
~ ››› traceroute wikipedia.org
traceroute to wikipedia.org (91.198.174.192), 64 hops max, 52 byte packets
 1  setup.ubnt.com (172.16.1.1)  2.646 ms  3.643 ms  1.250 ms
 2  * winroute.uni-koblenz.de (141.26.64.9)  9.476 ms *
 3  g-uni-ko-1.rlp-net.net (217.198.241.129)  2.817 ms  23.365 ms  21.333 ms
 4  g-hbf-ko-1.rlp-net.net (217.198.240.69)  8.481 ms  2.680 ms  5.767 ms
 5  g-hbf-mz-2.rlp-net.net (217.198.240.21)  3.766 ms  3.781 ms  4.305 ms
 6  g-interxion-1.rlp-net.net (217.198.240.13)  7.089 ms  12.138 ms  4.881 ms
 7  r1fra3.core.init7.net (80.81.192.67)  7.519 ms  4.139 ms  4.076 ms
 8  r1ams1.core.init7.net (77.109.128.154)  12.420 ms  14.985 ms  12.392 ms
 9  r1ams2.core.init7.net (77.109.128.146)  12.446 ms  14.863 ms  12.615 ms
10  * * *
11  * *^C
```

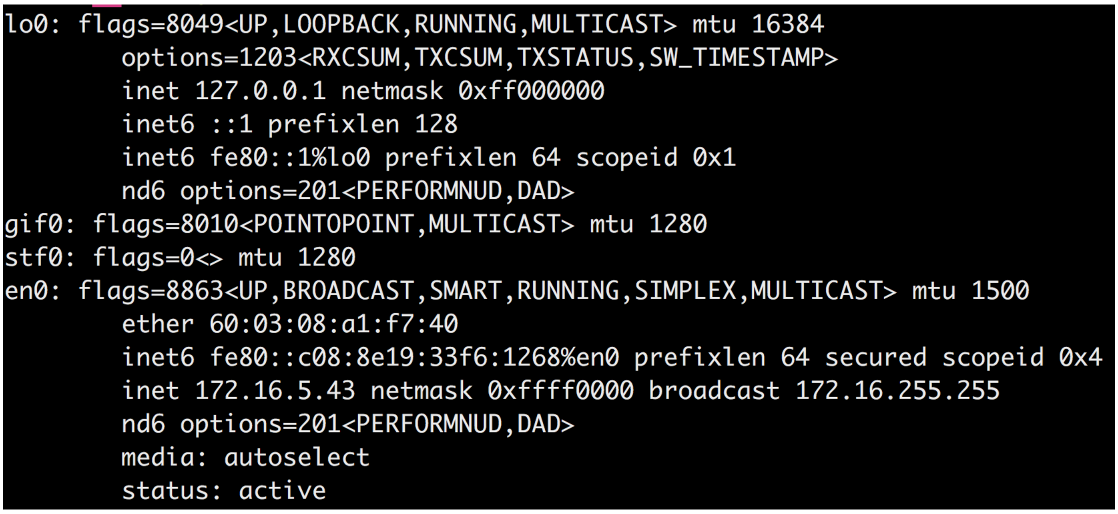
b) University network:
It took 9 Hops to get to the www.wikipedia.org server (traceroute www.wikipedia.org)



```
~ ››› traceroute www.wikipedia.org
traceroute to www.wikipedia.org (91.198.174.192), 64 hops max, 52 byte packets
 1  radius.uni-koblenz.de (172.16.12.1)  2.093 ms  1.096 ms  1.062 ms
 2  * winroute.uni-koblenz.de (141.26.64.9)  2.376 ms *
 3  g-uni-ko-1.rlp-net.net (217.198.241.129)  3.192 ms  2.693 ms  2.722 ms
 4  g-hbf-ko-2.rlp-net.net (217.198.247.69)  2.344 ms  2.283 ms  2.114 ms
 5  g-hbf-mz-1.rlp-net.net (217.198.240.105)  4.369 ms  4.118 ms  4.626 ms
 6  g-interxion-1.rlp-net.net (217.198.240.9)  3.335 ms  4.407 ms  3.360 ms
 7  r1fra3.core.init7.net (80.81.192.67)  4.025 ms  3.597 ms  3.440 ms
 8  r1ams1.core.init7.net (77.109.128.154)  11.745 ms  12.320 ms  11.974 ms
 9  r1ams2.core.init7.net (77.109.128.146)  11.961 ms  12.115 ms  11.955 ms
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
```

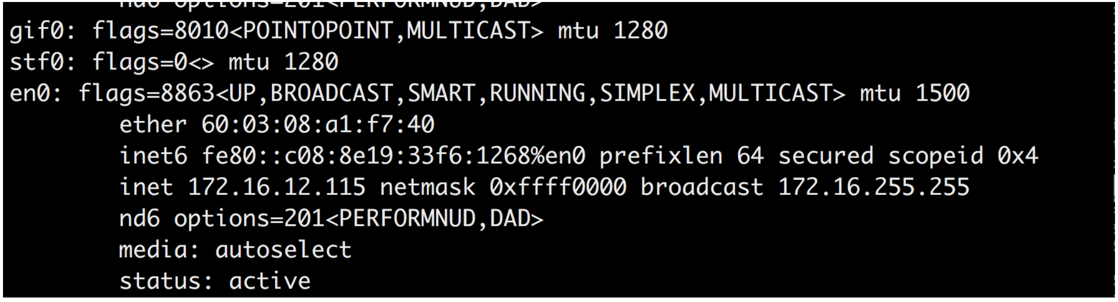
6. MAC address of the device that is acting as your network gateway

a) Home network (ifconfig)
The MAC address of the device is 60:03:08:a1:f7:40



```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 60:03:08:a1:f7:40
        inet6 fe80::c08:8e19:33f6:1268%en0 prefixlen 64 secured scopeid 0x4
        inet 172.16.5.43 netmask 0xffff0000 broadcast 172.16.255.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
```

b) University Network (ifconfig)
The MAC address of the device is 60:03:08:a1:f7:40



```
        nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 60:03:08:a1:f7:40
        inet6 fe80::c08:8e19:33f6:1268%en0 prefixlen 64 secured scopeid 0x4
        inet 172.16.12.115 netmask 0xffff0000 broadcast 172.16.255.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
```

4 Simple Python Programming (10 Points)

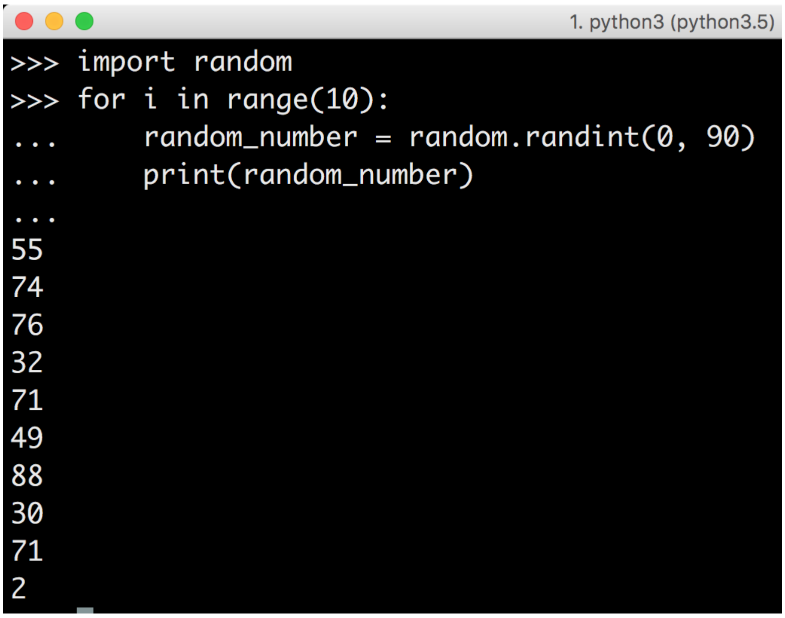
Write a simple python program that does the following:

1. Generate a random number sequence of 10 values between 0 to 90.
2. Perform `sine` and `cosine` operation on numbers generated.
3. Store the values in two different arrays named SIN & COSIN respectively.
4. Plot the values of SIN & COSIN in two different colors.
5. The plot should have labeled axes and legend.

Answers

1. Generate a random number sequence of 10 values between 0 to 90.

```
import random
for i in range(10):
        random_number = random.randint(0, 90)
        print(random_number)
```

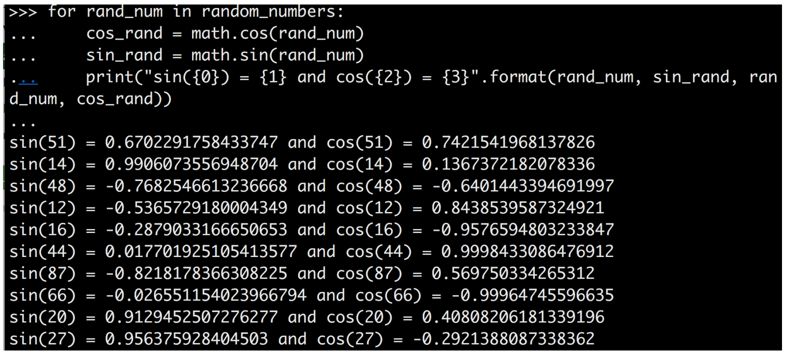


```
>>> import random
>>> for i in range(10):
...     random_number = random.randint(0, 90)
...     print(random_number)
...
55
74
76
32
71
49
88
30
71
2
```

2. Perform `sine` and `cosine` operation on numbers generated.

```
import random
import math
random_numbers = []
for i in range(10):
        random_number = random.randint(0, 90)
        random_numbers.append(random_number)
```

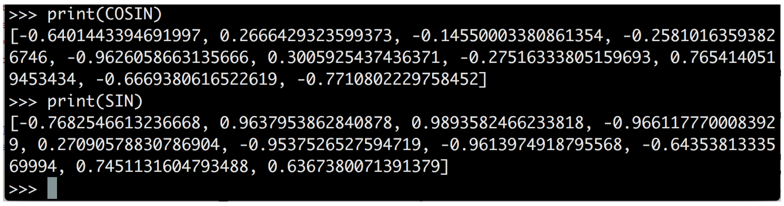
```

for rand_num in random_numbers:
        cos_rand = math.cos(rand_num)
        sin_rand = math.sin(rand_num)
        print("sin({0}) = {1} and cos({2}) = {3}".format(rand_num, sin_rand, rand_
```



```
>>> for rand_num in random_numbers:
...     cos_rand = math.cos(rand_num)
...     sin_rand = math.sin(rand_num)
...     print("sin({0}) = {1} and cos({2}) = {3}".format(rand_num, sin_rand, ran
d_num, cos_rand))
...
sin(51) = 0.6702291758433747 and cos(51) = 0.7421541968137826
sin(14) = 0.9906073556948704 and cos(14) = 0.1367372182078336
sin(48) = -0.7682546613236668 and cos(48) = -0.6401443394691997
sin(12) = -0.5365729180004349 and cos(12) = 0.8438539587324921
sin(16) = -0.2879033166650653 and cos(16) = -0.9576594803233847
sin(44) = 0.017701925105413577 and cos(44) = 0.9998433086476912
sin(87) = -0.8218178366308225 and cos(87) = 0.569750334265312
sin(66) = -0.026551154023966794 and cos(66) = -0.99964745596635
sin(20) = 0.9129452507276277 and cos(20) = 0.40808206181339196
sin(27) = 0.956375928404503 and cos(27) = -0.2921388087338362
```

3. Store the values in two different arrays named SIN & COSIN respectively.

```
import random
import math
random_numbers = []
COSIN = []
SIN = []
for i in range(10):
        random_number = random.randint(0, 90)
        random_numbers.append(random_number)

for rand_num in random_numbers:
        cos_rand = math.cos(rand_num)
        COSIN.append(cos_rand)
        sin_rand = math.sin(rand_num)
        SIN.append(sin_rand)
print(COSIN)
print(SIN)
```

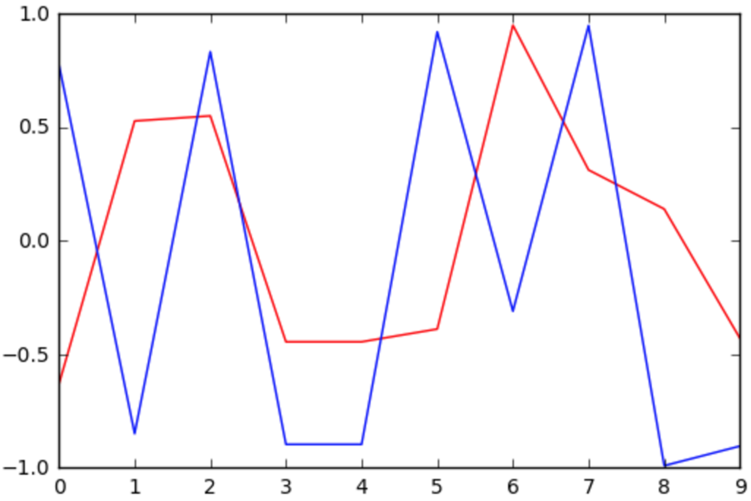


```
>>> print(COSIN)
[-0.6401443394691997, 0.2666429323599373, -0.14550003380861354, -0.2581016359382
6746, -0.9626058663135666, 0.3005925437436371, -0.27516333805159693, 0.765414051
9453434, -0.6669380616522619, -0.7710802229758452]
>>> print(SIN)
[-0.7682546613236668, 0.9637953862840878, 0.9893582466233818, -0.966117770008392
9, 0.27090578830786904, -0.9537526527594719, -0.9613974918795568, -0.64353813335
69994, 0.7451131604793488, 0.6367380071391379]
>>> 
```

4. Plot the values of SIN & COSIN in two different colors.

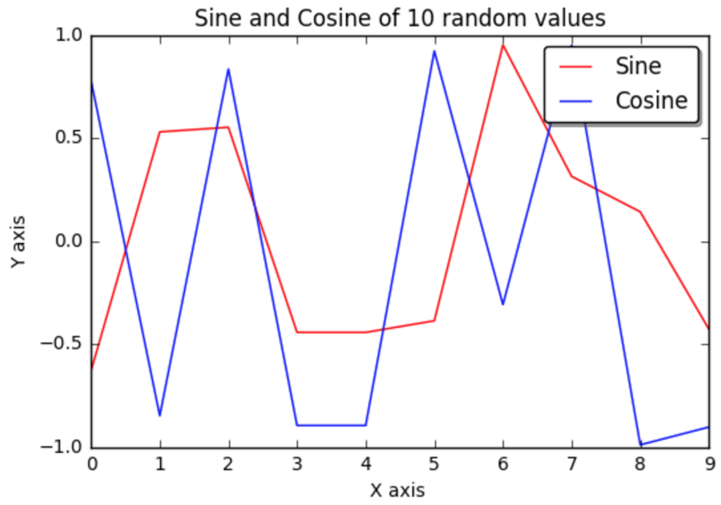
```
import random
import math
import matplotlib.pyplot as plt
```

```
random_numbers = []
COSIN = []
SIN = []
for i in range(10):
        random_number = random.randint(0, 90)
        random_numbers.append(random_number)

for rand_num in random_numbers:
        cos_rand = math.cos(rand_num)
        COSIN.append(cos_rand)
        sin_rand = math.sin(rand_num)
        SIN.append(sin_rand)


# Plotting
plt.plot(SIN, color="red", label="Sine")
plt.plot(COSIN, color="blue", label="Cosine")
plt.show()
```



4. The plot should have labeled axes and legend.

```
import random
import math
import matplotlib.pyplot as plt
random_numbers = []
COSIN = []
SIN = []
for i in range(10):
        random_number = random.randint(0, 90)
        random_numbers.append(random_number)

for rand_num in random_numbers:
        cos_rand = math.cos(rand_num)
        COSIN.append(cos_rand)
        sin_rand = math.sin(rand_num)
        SIN.append(sin_rand)

# Plotting
plt.plot(SIN, color="red", label="Sine")
plt.plot(COSIN, color="blue", label="Cosine")
plt.title('Sine and Cosine of 10 random values')
plt.xlabel('X axis ')
plt.ylabel('Y axis')
plt.legend(loc='upper right')
plt.legend( shadow=True, fancybox=True)
plt.show()
```



Important Notes

Submission

- Solutions have to be checked into the github repository. Use the directory name `groupname/assignment1/` in your group's repository.
- The name of the group and the names of all participating students must be listed on each submission.
- Solution format: all solutions as *one* PDF document. Programming code has to be submitted as Python code to the github repository. Upload *all* `.py` files of your program! Use `UTF-8` as the file encoding. *Other encodings will not be taken into account!*
- Check that your code compiles without errors.
- Make sure your code is formatted to be easy to read.
  - Make sure you code has consistent indentation.
  - Make sure you comment and document your code adequately in English.
  - Choose consistent and intuitive names for your identifiers.
- Do *not* use any accents, spaces or special characters in your filenames.

Acknowledgment

This latex template was created by Lukas Schmelzeisen for the tutorials of ”Web Information Retrieval”.