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: echo

Group Member: Hanadi Tamimi, Keya Kashem, Md Jakaria Nawaz

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. The type length has two bytes. If the hex value is less than 06 00 then it indicates the field is used as the size of the payload of the Ethernet frame. But if the hex value is more than 06 00 then the field is used to represent EtherType. Here it is 08 06 > 06 00. So it represents EtherType and 08 06 indicates Address Resolution Protocol (ARP).
4. The last one indicates the amount of data is being transferred. The range is 46-1500 bytes. In this case the data: 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 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

   The field before that indicates either EtherType (if HexValue > 0600) or size of the payload (if HexValue < 0600).

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. Cable length 20 m
   Network bandwidth 100MBps = 100 Million bits per second
   Or 1 bit per $(1/100 Million)$ seconds or 0.00000001 seconds or 10 nanoseconds.
   The speed of light = 299792458 m/s. (Data will pass in this speed as electromagnetic wave)
   In 1 sec data travel 299792458 m [in 100MBps Cable]
   So, in .00000001 sec 1 bit will travel $(299792458 * 0.00000001)m = 2.998$ m
   So, to travel 20 meters it will take $(20 * 10)/2.998 nanoseconds = 66.71$ nanoseconds

2. Cable length 20 m
   Network bandwidth 10MBps = 10 Million bits per second
   Or 1 bit per $(1/10 Million)$ seconds or 0.0000001 seconds or 100 nanoseconds.
   The speed of light = 299792458 m/s. (Data will pass in this speed as electromagnetic wave)
   In 1 sec data travel 299792458 m [in 100MBps Cable]
   So, in .0000001 sec 1 bit will travel $(299792458 * 0.0000001)m = 29.9792$ m
   So, to travel 20 meters it will take $(20 * 100)/29.979 nanoseconds = 66.71$ nanoseconds

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*

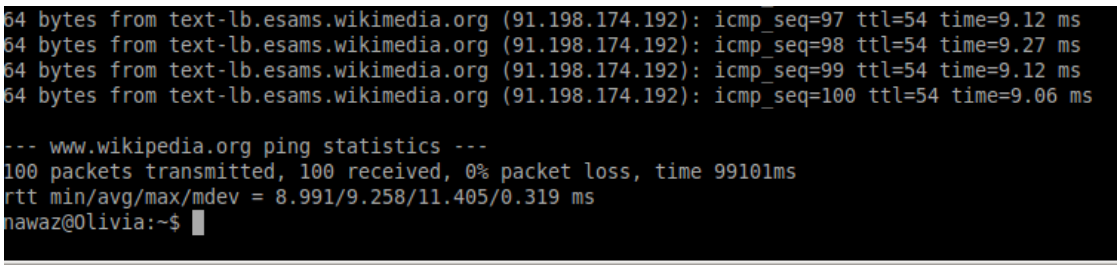
Consider a situation in which you need to check if `www.wikipedia.org` is reachable or not. Using the knowledge you gained above to find the following information:

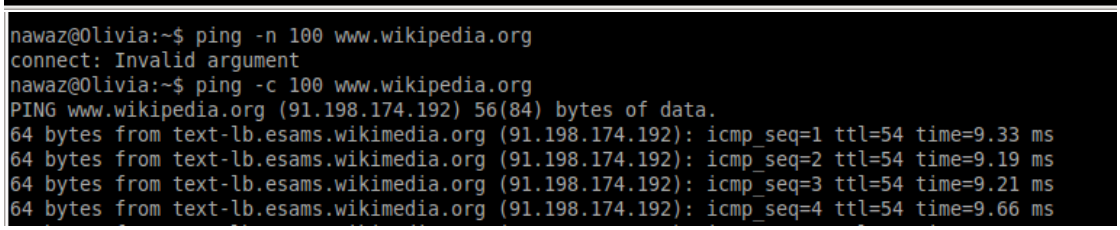
1. The *% packet loss* if at all it happened after sending 100 packets.
2. *Size* of the packet sent to *Wikipedia* server
3. *IP address* of your machine and the *Wikipedia* server
4. *Query Time* for DNS query of the above url.
5. Number of *Hops* in between your machine and the server
6. MAC address 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. Home: After sending 100 packets of data the percentage of packet loss is 0.(ping -c 100 www.wikipedia.org)





University: After sending 100 packets of data the percentage of packet loss is 0.(ping -n 100 www.wikipedia.org)

```
64 bytes from 91.198.174.192: icmp_seq=86 ttl=57 time=34.578 ms
64 bytes from 91.198.174.192: icmp_seq=87 ttl=57 time=33.884 ms
64 bytes from 91.198.174.192: icmp_seq=88 ttl=57 time=33.587 ms
64 bytes from 91.198.174.192: icmp_seq=89 ttl=57 time=33.543 ms
64 bytes from 91.198.174.192: icmp_seq=90 ttl=57 time=33.757 ms
64 bytes from 91.198.174.192: icmp_seq=91 ttl=57 time=44.633 ms
64 bytes from 91.198.174.192: icmp_seq=92 ttl=57 time=33.640 ms
64 bytes from 91.198.174.192: icmp_seq=93 ttl=57 time=32.156 ms
64 bytes from 91.198.174.192: icmp_seq=94 ttl=57 time=40.234 ms
64 bytes from 91.198.174.192: icmp_seq=95 ttl=57 time=37.974 ms
64 bytes from 91.198.174.192: icmp_seq=96 ttl=57 time=48.224 ms
64 bytes from 91.198.174.192: icmp_seq=97 ttl=57 time=47.954 ms
64 bytes from 91.198.174.192: icmp_seq=98 ttl=57 time=53.555 ms
64 bytes from 91.198.174.192: icmp_seq=99 ttl=57 time=32.721 ms

--- wikipedia.org ping statistics ---
100 packets transmitted, 100 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 31.997/40.871/103.653/9.791 ms
Keyas-MBP:~ keyak02$
```

2. Home: Size of the packet sent to wikipedia server is 56 bytes. (ping www.wikipedia.org)

```
nawaz@Olivia:~$ ping www.wikipedia.org
PING www.wikipedia.org (91.198.174.192) 56(84) bytes of data.
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=1 ttl=54 time=9.32 ms
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=2 ttl=54 time=9.32 ms
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=3 ttl=54 time=9.17 ms
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=4 ttl=54 time=10.1 ms
```

University: Size of the packet sent to wikipedia server is 56 bytes. (ping www.wikipedia.org)

```
Last login: Tue Nov  1 22:24:12 on ttys004
[Keyas-MacBook-Pro:~ keyak02$ 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=57 time=41.109 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=57 time=38.394 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=57 time=40.416 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=57 time=48.721 ms
64 bytes from 91.198.174.192: icmp_seq=4 ttl=57 time=39.238 ms
```

3. Home: IP address of machine - 172.16.5.57 . (ifconfig)

```
nawaz@Olivia:~$ /sbin/ifconfig
eth0      Link encap:Ethernet  HWaddr 1c:75:08:a1:1c:18
          inet addr:172.16.5.57  Bcast:172.16.255.255  Mask:255.255.0.0
          inet6 addr: fe80::1e75:8ff:fea1:1c18/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:22864964 errors:0 dropped:0 overruns:0 frame:1
          TX packets:12050918 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:26278945860 (26.2 GB)  TX bytes:2334608459 (2.3 GB)
          Interrupt:18
```

IP address of wikipedia server 97.198.174.192 . (ping www.wikipedia.org)

```
nawaz@Olivia:~$ ping www.wikipedia.org
PING www.wikipedia.org (91.198.174.192) 56(84) bytes of data.
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=1 ttl=54 time=9.32 ms
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=2 ttl=54 time=9.32 ms
```

University: IP address of machine - 192.168.178.51 . (ifconfig)

```
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether f4:5c:89:c9:8d:31
        inet6 fe80::f65c:89ff:fec9:8d31%en0 prefixlen 64 scopeid 0x4
        inet 192.168.178.51 netmask 0xffffff00 broadcast 192.168.178.255
        inet6 2a02:810b:400:3988:f65c:89ff:fec9:8d31 prefixlen 64 autoconf
        inet6 2a02:810b:400:3988:db5:92b5:a907:e5eb prefixlen 64 autoconf temporar
        nd6 options=1<PERFORMNUD>
        media: autoselect
        status: active
```

IP address of wikipedia server 97.198.174.192 . (ping www.wikipedia.org)

```
[Keyas-MBP:~ keyak02$ ping wikipedia.org
 PING wikipedia.org (91.198.174.192): 56 data bytes
 64 bytes from 91.198.174.192: icmp_seq=0 ttl=57 time=39.720 ms
 64 bytes from 91.198.174.192: icmp_seq=1 ttl=57 time=62.767 ms
 64 bytes from 91.198.174.192: icmp_seq=2 ttl=57 time=33.983 ms
 64 bytes from 91.198.174.192: icmp_seq=3 ttl=57 time=35.502 ms
 64 bytes from 91.198.174.192: icmp_seq=4 ttl=57 time=42.682 ms
 64 bytes from 91.198.174.192: icmp_seq=5 ttl=57 time=33.351 ms
 64 bytes from 91.198.174.192: icmp_seq=6 ttl=57 time=34.468 ms
 64 bytes from 91.198.174.192: icmp_seq=7 ttl=57 time=35.228 ms
 64 bytes from 91.198.174.192: icmp_seq=8 ttl=57 time=38.878 ms
 64 bytes from 91.198.174.192: icmp_seq=9 ttl=57 time=279.730 ms
 64 bytes from 91.198.174.192: icmp_seq=10 ttl=57 time=237.235 ms
 64 bytes from 91.198.174.192: icmp_seq=11 ttl=57 time=48.849 ms
 64 bytes from 91.198.174.192: icmp_seq=12 ttl=57 time=148.477 ms
 64 bytes from 91.198.174.192: icmp_seq=13 ttl=57 time=36.968 ms
```

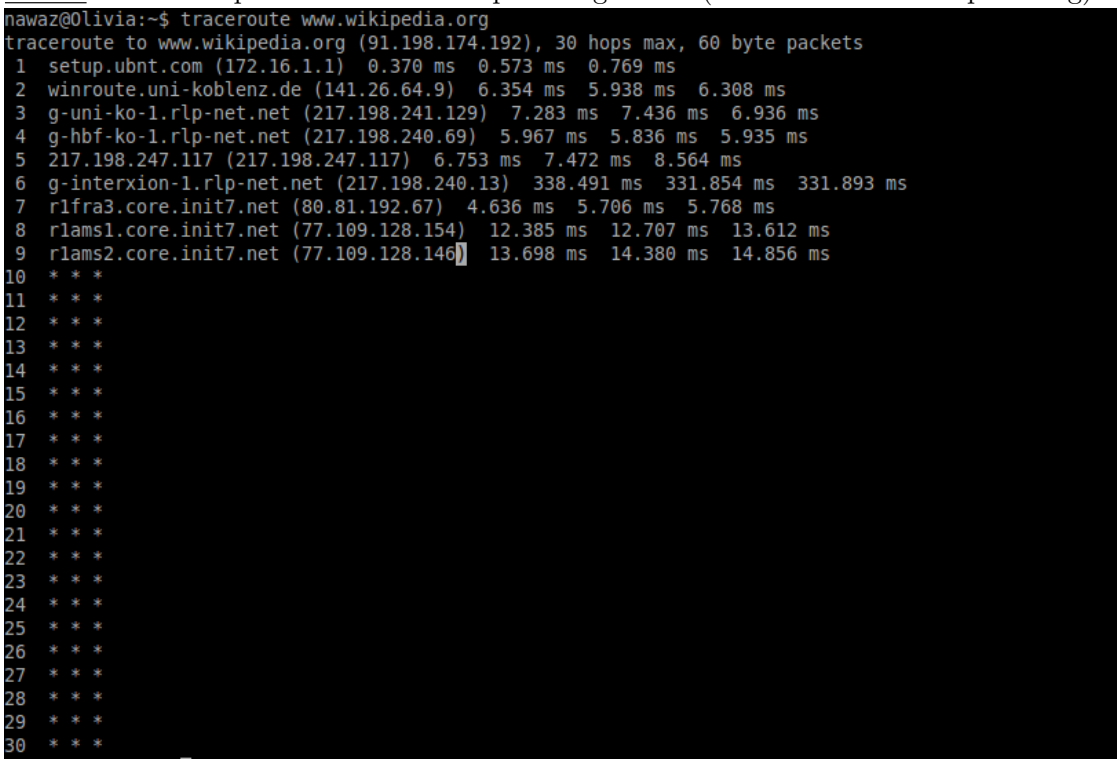
4. Home: Query time for DNS query of www.wikipedia.org - 13 msec . (dig www.wikipedia.org)

```
nawaz@Olivia:~$ dig wikipedia.org

; <<>> DiG 9.9.5-3ubuntu0.9-Ubuntu <<>> wikipedia.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9845
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 4

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;wikipedia.org.                 IN      A

;; ANSWER SECTION:
wikipedia.org.          600     IN      A       91.198.174.192

;; AUTHORITY SECTION:
wikipedia.org.          5076    IN      NS      ns2.wikimedia.org.
wikipedia.org.          5076    IN      NS      ns0.wikimedia.org.
wikipedia.org.          5076    IN      NS      ns1.wikimedia.org.

;; ADDITIONAL SECTION:
ns0.wikimedia.org.      1377    IN      A       208.80.154.238
ns1.wikimedia.org.      1377    IN      A       208.80.153.231
ns2.wikimedia.org.      1377    IN      A       91.198.174.239

;; Query time: 13 msec
;; SERVER: 127.0.1.1#53(127.0.1.1)
;; WHEN: Tue Nov 01 15:34:14 CET 2016
;; MSG SIZE  rcvd: 170
```

University: Query time for DNS query of www.wikipedia.org - 41 msec . (dig www.wikipedia.org)

```
Keyas-MacBook-Pro:~ keyak02$ dig wikipedia.org

; <<>> DiG 9.8.3-P1 <<>> wikipedia.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19384
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

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

;; ANSWER SECTION:
wikipedia.org.          319     IN      A       91.198.174.192

;; Query time: 41 msec
;; SERVER: 192.168.178.1#53(192.168.178.1)
;; WHEN: Tue Nov  1 23:06:27 2016
;; MSG SIZE  rcvd: 47
```

5. Home: It took 9 hops to reach www.wikipedia.org server. (traceroute www.wikipedia.org)



University: It took 6 hops to reach www.wikipedia.org server. (traceroute www.wikipedia.org)

```
Keyas-MacBook-Pro:~ keyak02$ traceroute wikipedia.org
traceroute to wikipedia.org (91.198.174.192), 64 hops max, 52 byte packets
 1  fritz.box (192.168.178.1)  8.815 ms  2.047 ms  1.275 ms
 2  83-169-183-67-isp.superkabel.de (83.169.183.67)  24.754 ms  18.608 ms  27.372 ms
 3  ip5886c969.dynamic.kabel-deutschland.de (88.134.201.105)  17.594 ms  19.791 ms  20.165 ms
 4  ip5886caf0.dynamic.kabel-deutschland.de (88.134.202.240)  18.998 ms
    ip5886eb58.dynamic.kabel-deutschland.de (88.134.235.88)  17.382 ms  18.255 ms
 5  ip5886eb0e.dynamic.kabel-deutschland.de (88.134.235.14)  34.156 ms
    ip5886ca35.dynamic.kabel-deutschland.de (88.134.202.53)  31.317 ms  28.469 ms
 6  ip5886cac1.dynamic.kabel-deutschland.de (88.134.202.193)  35.482 ms  33.276 ms  42.471 ms
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
```

6. Home: MAC address of network gateway 04:18:d6:83:a9:83. (route -n, arp -n)

```
nawaz@Olivia:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         172.16.1.1      0.0.0.0         UG    0      0        0 eth0
10.42.0.0       0.0.0.0         255.255.255.0   U     9      0        0 wlan0
172.16.0.0      0.0.0.0         255.255.0.0     U     1      0        0 eth0
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
nawaz@Olivia:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
172.16.1.1               ether   04:18:d6:83:a9:83   C                     eth0
10.42.0.32                       (incomplete)                              wlan0
172.16.5.125             ether   00:24:fe:53:cb:05   C                     eth0
10.42.0.41               ether   48:5a:3f:5f:4f:60   C                     wlan0
172.16.2.178                     (incomplete)                              eth0
172.16.5.61                      (incomplete)                              eth0
10.42.0.17                       (incomplete)                              wlan0
172.16.3.56              ether   fc:f1:52:bb:61:1c   C                     eth0
10.42.0.78                       (incomplete)                              wlan0
172.16.5.155             ether   c0:11:73:1d:04:6d   C                     eth0
10.42.0.84                       (incomplete)                              wlan0
```

University: MAC address of network gateway 34:31:c4:7d:65:8a. (netstat -nr, for macintosh device)

```
       arp -f filename
Keyas-MacBook-Pro:~ keyak02$ netstat -nr
Routing tables

Internet:
Destination        Gateway            Flags        Refs      Use   Netif Expire
default            192.168.178.1      UGSc          285        0     en0
127                127.0.0.1          UCS             1        0     lo0
127.0.0.1          127.0.0.1          UH              6   280971     lo0
169.254            link#4             UCS             1        0     en0
192.168.178        link#4             UCS             4        0     en0
192.168.178.1/32   link#4             UCS             6        0     en0
192.168.178.1      34:31:c4:7d:65:8a  UHLWIir        96      197     en0   1168
192.168.178.42     a4:77:33:b9:33:10  UHLWIi          3     1835     en0      1
192.168.178.51/32  link#4             UCS             1        0     en0
192.168.178.254    34:31:c4:7d:65:8b  UHLWIi          2       56     en0
192.168.178.255    link#4             UHLWbI          1       40     en0
224.0.0            link#4             UmCS            2        0     en0
224.0.0.251        1:0:5e:0:0:fb      UHmLWI          1        0     en0
255.255.255.255/32 link#4             UCS             1        0     en0

Internet6:
Destination                             Gateway                         Flags         Netif Expire
default                                 fe80::3631:c4ff:fe7d:658a%en0   UGc             en0
::1                                     ::1                             UHL             lo0
2a02:810b:400:3988::/64                 link#4                          UC              en0
2a02:810b:400:3988:db5:92b5:a907:e5eb   f4:5c:89:c9:8d:31               UHL             lo0
2a02:810b:400:3988:f65c:89ff:fec9:8d31  f4:5c:89:c9:8d:31               UHL             lo0
fe80::%lo0/64                           fe80::1%lo0                     UcI             lo0
fe80::1%lo0                             link#1                          UHLI            lo0
fe80::%en0/64                           link#4                          UCI             en0
fe80::3631:c4ff:fe7d:658a%en0           34:31:c4:7d:65:8a               UHLWIir         en0
fe80::f65c:89ff:fec9:8d31%en0           f4:5c:89:c9:8d:31               UHLI            lo0
fe80::%awdl0/64                         link#8                          UCI           awdl0
fe80::cc33:30ff:feec:fe59%awdl0         ce:33:30:ec:fe:59               UHLI            lo0
ff01::%lo0/32                           ::1                             UmCI            lo0
ff01::%en0/32                           link#4                          UmCI            en0
ff01::%awdl0/32                         link#8                          UmCI          awdl0
ff02::%lo0/32                           ::1                             UmCI            lo0
ff02::%en0/32                           link#4                          UmCI            en0
ff02::%awdl0/32                         link#8                          UmCI          awdl0
Keyas-MacBook-Pro:~ keyak02$
```

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 import random
2 import math
3 import matplotlib.pyplot as plt
4 import matplotlib.patches as mpatches
5 a = []
6 SIN = []
7 COSIN = []
8 for i in xrange(10):
9     a.append('%04.3f' % random.uniform(0,90))
10 for item in a:
11    COSIN.append(math.cos(float(item)))
12    SIN.append(math.sin(float(item)))
13 plt.plot(SIN,'-b', label='Sin')
14 plt.plot(COSIN,'-r', label='Cosine')
15 plt.axis()
16 plt.ylabel('y axis')
17 plt.xlabel('x axis')
18 plt.legend(loc='lower right')
19 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".