In-Class Work: Python — Repetitions & Arrays & Lists, and Functions And getting files to linuxremote

Solutions to Inclass-Work 10.-12. are in `~kvollmay/share.dir/inclass2025.dir/classpython10.py` etc.

12.2. Repetitions Sample Program

To practice how repetition commands work, copy into your working directory the sample program

```
~kvollmay/share.dir/pythonsamples.dir/sample_repetitions.py
```

Look at this program and before running this program, lookup online the used commands. Still before running this program, predict what exactly the program will print on the screen when you run the program. After your prediction, run the program and compare with your prediction.

13. I will show the program `~kvollmay/share.dir/inclass2025.dir/classpython13.py` This program does the following task:

Write a program that prints Sign-Up times for Mo, Wed, Fr and on each day 10:0, 10:10, 10:20,... 10:50, 11:0,11:10,... 11:50. So the first few and last few lines of out put should be

```
Mo 10 : 0
Mo 10 : 10
...
Fr 11 : 40
Fr 11 : 50
```

14. Write a program that reads in an integer N and a float $r < 1$, and determines and prints out $\sum_{k=0}^{N} r^k$ and also for comparison $\frac{\left(1-r^{N+1}\right)}{(1-r)}$. Test your program for several values of N and r.

16. Lists & Arrays Intro: Containers

For examples how to use arrays copy into your working directory the sample program

```
~kvollmay/share.dir/pythonsamples.dir/sample_arrays.py
```

When you run the script it reads in two files, so also do following copy commands

```
cp ~kvollmay/share.dir/pythonsamples.dir/in1.dat ./
cp ~kvollmay/share.dir/pythonsamples.dir/in2.dat ./
```

To learn more about lists and arrays, scan this sample file and then read in Newman as described in the following steps.

17. Lists

Start reading at the bottom of page 46 §2.4. Continue reading §2.4.1 and as practice try for page 48 the following commands:

```
r=[1,1,2,3,8]
print(r)
```

and also

```
x=1.0
y=1.5
z=-2.2
r=[x,y,z]
print(r)
y=2.0
print(r)
r=[x,y,z]
print(r)
```

On page 49 and all following pages of this section, replace the `math` package with the `numpy` package the same way as we did previously in this course. For example `sqrt(3.0)` is instead `np.sqrt(3.0)` and at the top of the program you need `import numpy as np`
In additon try the commands

```
a=[3,2.5,4,"hi",-3]
print(a)
print(a[0],a[1],a[3])
a[1]=7.3
print(a)
```

and for page 51 try

```
import numpy as np
r=[1.0,1.5,2.2]
logr=list(map(np.log,r))
print(logr)
```

and for pages 51 and 52 test

```
r=[1.0,1.5,2.2,9.1]
print(r)
r.append(2.8)
print(r)
r.pop()
print(r)
r.pop(1)
print(r)
print(r[2])
```

18. Arrays

Read §2.4.2 pages 53 and 54. We will also use numpy, but import differently so the commands on page 54 are replaced by

```
import numpy as np
a=np.zeros(4,float)
print(a)
```

and try also

```
b=np.zeros([2,3],int)
print(b)
```

for pages 55 and 56 try

```
d=np.array([[1,2,3],[4,5,6]],int)
print(d)
d[1,2]=30
print(d)
```

Continue reading, so read §2.4.3. Notice that `loadtxt` allows us to read data from a file, so we do not have to type in by hand each value. Get a feel for how it works, by using in your linux terminal `gedit` (or any other editor you like) to make the file named `values.txt` with the content as the four last lines on page 57. Then you can use a python script which includes the line `import numpy as np` to test the python commands

```
e=np.loadtxt("values.txt",float)
print(e)
```

In case this made you curious about how to also write into a file of a specified name, you may have a second look at `~kvollmay/share.dir/pythonsamples.dir/sample_arrays.py`

19. Traffic Model (application of §2.4 & §2.5)

Copy the file

`~kvollmay/share.dir/inclass2025.dir/road.dat`

into your working directory. The file contains 100 integers describing a road of 100 lattice sites at a certain time. `-1` means an empty site and a number ≥ 0 means a car with the corresponding speed in mi/h.

19a. Read in this file, so that you have an array with your values. Check with a print command that you succeeded. (**Hint:** use from `~kvollmay/share.dir/pythonsamples.dir/sample_arrays.py` the command `np.loadtxt`)

19b. How many cars N are on this road?

19c. What is the average velocity

$$\langle v \rangle = \frac{1}{N} \sum_{i=1}^{N} v_i$$

where v_i is the velocity of car i and there are N cars.

19d. What is the largest velocity ?

20. Fun with Matrix

20a. Write a program that sets an array to be as follows:

0	0	0	0	0
0	1	1	1	0
0	0	1	0	0
0	0	0	0	0
0	0	0	0	0

Check your program with a print command.

20b. Now change the printing so that there are no square brackets being printed. To avoid getting a new line for each lattice value, use that for example

```
print(7,end=" ")
```

does end with the space character and not with the new line character.

20c. Define the length of the array, i.e. 5, as variable you set at the beginning of your program. This will allow us later to change the size of our lattice easily.

20d. (Tool for Fun) You can look at this array graphically by adding to your python script the following command at the beginning

```
import pylab as pl
```

and then after you have set your array the following two lines (assuming that your array is called `lat`)

```
pl.imshow(lat,interpolation='nearest')
pl.show()
```

For more information Newman describes this tool in §3.3.

Only If Time

21. User-Defined Functions

21a. Start reading in Newman's book §2.6 (page 75) and as practice for the understanding of page 76 and the first half of page 77, test the commands

```
def factorial(n):
  f = 1.0
  for k in range(1,n+1):
    f *= k
  return f

print(factorial(10))
```

21b. Next add to your program that n is read from input and then $n!$ is printed.

21c. Next let's combine this user-defined function with the repetition tools from last class. Write a program which prints out n and $n!$ for $n = 1, 4, 7, 10, 13, \ldots, 22$.

21d. Continue reading on page 77, and include in your program the user-defined function `distance` . If you want to use `numpy` make sure to include in your program
`import numpy as np`
and now add to your program (or write a small new python script) with the lines

```
def distance  (r,theta,z):
  x = r*np.cos(theta)
  y = r*np.sin(theta)
  d= np.sqrt(x**2+y**2+z**2)
  print("x ",x," y= ",y," d=",d)
  return d
```

and test your program for the case of $r = 2.0, \theta = 0.5\pi, z = 1.0$ and check whether you get the expected value.

22. Good Programming Style

Read carefully §2.7. I could not agree more with each of Newman's rules.

Getting files to linuxremote

Here a few tools to get files transferred to linuxremote and back to other filesystem:

1. Using Email

When you are using FastX, then you can open a web browser within the FastX window. From the web browser you get to your email and so that allows you to receive a file you have sent to yourself via email. As usual, you can download the file. Within FastX is will most likely and up in /Downloads/ and that means you can then copy the file in the FastX terminal window as usual, so like `cp ~/Downloads/file1.pdf ~/share.dir/` which would copy the file1.pdf into your `~/share.dir/`

2. If on Bucknell netspace via Windows Environment

1. in File Explorer right-click on "This PC" and you will see the option to map a network drive.
2. Left-click Map Network Drive
3. In the Drive: box, select any available drive letter. (Default has worked in the past for me.)
4. In the Folder: box, type `\\unixspace\linux-username.$` (where username is your real Bucknell username)
5. Optionally check the Reconnect at logon button if you wish to have this drive mapped each time you logon to this machine.
6. Click the Finish button.

That gets you your linuxspace added to the File Explorer and so you can do regular copying within the windows environment.