competency_intro_python

November 4, 2023

[1]:
```python
import numpy
import math
import matplotlib.pyplot as plt
```

[2]:
```python
#Pierce Zhang, CMOR220, Fall 2023
#competency_intro_python.ipynb
#Answers the questions to the intro to Python competency
#Last modified: November 4, 2023
```

[3]:
```python
# Generates several numpy arrays as specified in the competency problem 1 parts␣
↪(a) - (e)
a = numpy.arange(11, 50, 2)
print("a = \n",a)

b = numpy.arange(0,9).reshape((3,3))
print("b = \n",b)

c = numpy.linspace(0,2,30)
print("c = \n",c)

d = numpy.random.rand(25)
print("d = \n",d)

e = numpy.random.rand(25,25)
print("e = \n",e)
```

```
a =
 [11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49]
b =
 [[0 1 2]
 [3 4 5]
 [6 7 8]]
c =
 [0.         0.06896552 0.13793103 0.20689655 0.27586207 0.34482759
 0.4137931  0.48275862 0.55172414 0.62068966 0.68965517 0.75862069
 0.82758621 0.89655172 0.96551724 1.03448276 1.10344828 1.17241379
 1.24137931 1.31034483 1.37931034 1.44827586 1.51724138 1.5862069
 1.65517241 1.72413793 1.79310345 1.86206897 1.93103448 2.        ]
```

```
d =
 [0.02776508 0.33774725 0.85370803 0.99866826 0.29467043 0.24943983
 0.56657121 0.26350579 0.67958869 0.37660111 0.53503673 0.28238488
 0.39446486 0.0848489  0.37772193 0.72465804 0.09219974 0.14724914
 0.33082595 0.25181089 0.84433618 0.97869846 0.35673888 0.82797515
 0.95422865]
e =
 [[4.99100777e-01 7.00620168e-01 4.56362185e-01 7.88301353e-01
  9.40787391e-01 6.31403494e-01 7.55675193e-01 5.12093127e-01
  9.64402216e-01 2.23625672e-01 5.13120054e-01 8.32758994e-01
  1.91139094e-01 2.49661425e-01 4.15223088e-01 7.20221819e-01
  3.15154173e-01 5.15652550e-02 8.39878357e-01 9.10096262e-01
  4.95257307e-01 1.62563177e-01 7.97650311e-01 4.20515869e-02
  4.94071152e-02]
 [9.43297157e-01 7.22199912e-02 4.26740746e-02 6.80481379e-01
  3.76414332e-01 2.24462045e-02 2.70683410e-01 2.49878793e-01
  6.42915613e-01 2.18579454e-01 8.27110491e-01 5.59788245e-01
  9.06726844e-01 6.15737156e-01 7.49112060e-01 5.02989157e-01
  1.55983631e-01 7.58988524e-02 6.14874935e-01 5.69434746e-01
  7.84519698e-01 2.25866634e-01 5.75523327e-01 9.88002864e-01
  4.59449438e-01]
 [3.13731960e-01 4.72360541e-01 1.11570153e-02 3.08903933e-01
  2.39428950e-01 5.36246329e-01 8.03279973e-01 9.33695113e-01
  9.52906406e-01 8.54217993e-04 3.69586459e-01 8.79942269e-01
  7.64905346e-01 5.28742212e-01 6.46322911e-01 8.53174742e-01
  5.51376050e-01 2.61243266e-01 8.37676701e-01 1.42176015e-01
  8.27078808e-01 5.64541158e-01 1.02462670e-01 3.42385192e-01
  6.85465983e-01]
 [4.97811672e-01 4.30323927e-01 7.24496768e-01 1.67061569e-01
  9.81960847e-01 1.55381804e-01 6.75280758e-01 2.53311648e-01
  2.25292233e-01 4.69987219e-01 8.65026836e-01 1.26536867e-01
  4.29251936e-01 2.34394380e-01 6.26514963e-01 5.90383723e-02
  1.30773176e-02 7.86962741e-01 3.32509034e-01 3.58432666e-01
  9.87843541e-01 5.14920225e-01 4.36621377e-01 8.50493523e-01
  8.39185657e-01]
 [6.35348378e-01 5.78713481e-01 3.24902789e-01 5.31322016e-01
  1.51216953e-01 1.22266631e-01 2.77837341e-02 3.46153363e-01
  4.96473436e-01 5.19777346e-02 2.72388683e-01 6.82959748e-01
  6.00734344e-02 2.97896286e-01 7.16013449e-01 5.97950176e-02
  9.08656553e-01 2.69490262e-01 7.76098606e-01 7.22974477e-01
  4.05685049e-01 6.78263551e-01 3.03573178e-01 9.86876141e-01
  6.02864840e-01]
 [1.56811967e-01 1.32539908e-01 2.56854298e-01 4.02356019e-01
  9.02292821e-01 4.98424085e-01 6.08696985e-03 7.08852843e-01
  5.60983183e-02 8.58636640e-01 1.09218506e-01 3.17435581e-01
  8.25762997e-01 1.73666742e-01 9.31825441e-01 6.81535559e-01
  9.82816192e-01 7.13289269e-01 3.22984370e-01 5.99227780e-01
  2.01205138e-01 9.00882187e-01 7.08184647e-01 4.41247137e-01
```

```
 5.03829956e-02]
[2.52747232e-01 9.57907783e-01 8.21170255e-01 3.14613076e-01
 6.18352674e-01 4.44475222e-01 6.94139666e-01 3.77741437e-01
 9.24537533e-01 7.81463538e-01 4.91992927e-01 3.04019684e-01
 9.73381219e-01 4.77721773e-01 6.62012045e-01 4.62476887e-01
 8.57901343e-01 3.85017609e-01 2.14623105e-01 5.24760618e-01
 2.87134404e-01 4.86841700e-01 5.95223926e-01 7.18098509e-01
 3.26692336e-01]
[2.09018612e-02 9.64612682e-01 1.63714288e-01 2.56898382e-01
 7.09422629e-01 2.32105618e-01 3.96959148e-01 7.42308092e-01
 6.91854057e-01 1.50205817e-01 8.28967754e-01 9.10237504e-01
 2.62260536e-02 7.80730903e-01 3.60383623e-01 6.13695250e-01
 9.20572304e-01 1.92014536e-01 1.64131944e-01 3.71849333e-01
 2.48229643e-01 1.82320469e-01 8.75424011e-01 3.25706816e-01
 4.13383475e-01]
[2.23020222e-01 9.97550144e-01 6.91444706e-01 1.74007434e-02
 7.78068290e-01 4.27038359e-01 1.18339335e-02 4.39092199e-01
 7.09504254e-01 1.17418488e-01 9.54810215e-01 4.05916625e-01
 4.10323047e-01 4.00767652e-01 3.28546930e-01 1.14941954e-01
 3.45096801e-01 7.30968527e-02 4.07335041e-01 8.81004563e-02
 4.21909621e-01 1.07583890e-01 7.41752484e-02 9.34106934e-01
 9.95905680e-02]
[7.53388715e-01 1.90273397e-01 4.68877748e-01 7.11266995e-01
 5.99383884e-01 4.03411168e-01 6.56535398e-01 3.53303757e-01
 9.82186482e-01 9.84921074e-03 6.80499657e-01 2.87882727e-01
 2.78256117e-01 8.69833006e-01 4.12308414e-01 6.61863152e-01
 6.37462852e-01 2.34994791e-01 8.02233672e-01 8.41101761e-01
 4.09662663e-01 3.54818014e-01 3.69164683e-01 7.74283613e-01
 4.02870893e-01]
[7.57473657e-01 5.81870042e-02 7.70767877e-02 1.80798099e-02
 6.80660693e-01 4.86016322e-01 1.20662258e-01 8.67223469e-02
 2.75843864e-01 5.39827616e-01 6.55499156e-01 3.77984031e-01
 2.76743792e-01 4.41976720e-02 3.53043988e-01 6.90983238e-01
 5.47862747e-01 2.23982978e-01 8.95176025e-01 9.91173121e-01
 9.31654923e-01 7.44047617e-01 7.90283618e-02 6.57673115e-02
 7.45629688e-01]
[5.92754762e-02 5.69723266e-01 8.74179142e-02 3.75461036e-01
 2.19691714e-01 6.23857568e-01 9.87662036e-01 7.90015760e-01
 2.30018987e-01 4.01153429e-01 6.03701447e-01 3.03722974e-01
 4.40173027e-01 2.30784502e-01 1.24009460e-03 2.41300932e-01
 8.34813484e-01 5.37589152e-01 4.21959198e-01 8.50438165e-01
 5.48235690e-01 4.19530752e-01 7.38063458e-01 6.94948419e-01
 8.22323949e-01]
[8.29667779e-01 9.36234598e-01 7.88262115e-01 5.98076019e-01
 8.21937077e-01 8.33567005e-01 1.64441935e-01 8.24408268e-01
 5.00663979e-01 8.63198185e-01 5.04904223e-01 7.15779756e-01
 2.29142864e-01 2.22387475e-01 1.58204096e-01 5.06290936e-01
 9.77730706e-01 6.48726329e-01 1.96537602e-01 6.34788702e-01
```

8.79580584e-01 7.75295013e-01 8.34275258e-01 1.63918424e-01
4.97082531e-02]
[9.58005095e-01 5.59810002e-01 4.96191706e-01 9.75943307e-01
1.87155675e-01 7.62345741e-01 6.35982751e-01 4.01629759e-01
8.36097214e-01 4.90201257e-01 9.47459253e-01 2.27659107e-01
4.42470590e-01 7.47621296e-01 2.26763090e-01 8.58138419e-01
7.48435814e-01 1.14389173e-02 3.42441097e-02 2.10676547e-01
5.47540451e-01 7.31694794e-01 5.12388587e-01 9.31988076e-01
2.91076905e-01]
[5.59694205e-01 1.01503791e-02 8.39682952e-01 2.33490720e-01
1.48174897e-02 4.70440548e-01 8.89609258e-01 3.43261118e-01
5.54596125e-01 2.66970544e-01 6.56174143e-01 9.19477555e-01
2.16665424e-04 5.27216065e-01 1.15059411e-01 9.48909805e-01
8.59900987e-01 6.87293297e-01 2.12582157e-01 3.37918185e-01
7.62004245e-02 3.54335026e-01 9.83632566e-01 5.31721868e-01
8.10098877e-01]
[3.17257716e-01 2.52637198e-01 4.53135401e-01 5.63347036e-02
6.21616464e-02 9.03853617e-01 5.95406832e-01 1.59341092e-02
6.53701783e-01 7.33870256e-01 1.10269120e-01 7.34209774e-01
1.83266495e-03 1.99556240e-01 8.64537366e-01 1.59494529e-01
6.30829918e-01 6.96704779e-01 3.94267043e-01 9.56440808e-01
1.39572054e-01 1.49514559e-01 4.33178135e-01 4.22950493e-01
1.01000048e-03]
[8.36261275e-01 2.38429886e-01 9.73520216e-01 6.47986350e-01
8.86499945e-01 2.13794829e-01 7.15487153e-01 5.68732083e-01
7.16172927e-01 4.06874786e-01 3.94654945e-01 5.07287750e-02
7.73673148e-01 4.16762570e-01 7.90248858e-01 8.79993581e-01
1.36991088e-01 4.98923066e-01 6.80856900e-01 3.75031208e-01
1.34173632e-01 8.90712624e-01 8.83747340e-01 5.31361691e-01
6.36399708e-01]
[1.51371967e-01 6.82723631e-01 1.67139019e-02 7.29607468e-01
7.50906727e-02 5.72775168e-01 7.65441025e-01 5.12505217e-01
1.36425655e-01 8.93402417e-01 9.85418017e-01 2.18827416e-01
4.53304215e-01 6.93753315e-01 8.78474566e-01 3.82942893e-01
6.01410937e-01 1.58268601e-01 5.46583909e-01 3.42286517e-01
3.74201836e-01 8.09759234e-01 8.31074849e-01 6.55664440e-01
5.42253559e-01]
[5.49340186e-02 8.26965534e-01 4.77089575e-01 4.41101499e-01
1.15168340e-01 6.90886080e-01 5.88310549e-01 4.88934809e-02
8.58633824e-01 7.32915219e-01 9.13151358e-01 9.33660744e-01
8.64545524e-01 4.93958673e-01 2.12296812e-01 7.69252257e-01
5.06954629e-01 9.00073306e-01 4.07301836e-01 9.95212893e-01
4.18912450e-01 8.39456750e-01 6.21993315e-01 5.13524980e-01
6.61503884e-01]
[8.96446590e-01 7.45906705e-02 2.70162465e-02 1.92147167e-02
5.83939076e-02 5.87958164e-01 3.03599733e-01 1.51044898e-01
6.37071954e-01 8.93687185e-01 1.76862480e-01 1.35019984e-01
8.89724226e-03 4.03627679e-01 8.00859838e-01 9.12436829e-01

```
 1.94508267e-01 3.94261182e-02 1.68568451e-01 2.89396314e-01
 3.02585878e-01 9.38948312e-01 1.12182364e-01 6.83868308e-01
 1.51348136e-01]
[6.64137621e-01 4.44984368e-01 8.68121898e-01 8.36506035e-01
 5.68346655e-01 7.35752326e-02 9.46623523e-01 1.09669353e-01
 4.92511971e-01 6.95590699e-01 6.44281060e-01 7.82945650e-01
 9.81093019e-01 6.86334019e-01 6.93994270e-01 5.22727913e-01
 3.06239669e-01 8.34309558e-01 1.01044265e-01 4.75179629e-01
 6.07555411e-02 6.01335598e-01 3.43318368e-01 9.20518030e-01
 2.16783010e-01]
[1.07091135e-01 8.57821906e-01 1.76875559e-01 2.77906789e-01
 9.88275575e-02 4.78019427e-01 2.45979815e-01 9.43912500e-01
 4.50245305e-01 7.51428383e-01 5.73428872e-01 6.01360307e-01
 4.97349200e-01 6.90664981e-01 2.20036003e-01 8.91415171e-01
 6.46302585e-02 2.32816324e-01 2.40057651e-01 8.32979159e-01
 9.02936362e-01 4.30291636e-01 2.46197899e-01 4.28732641e-01
 7.71106635e-01]
[9.37424137e-01 6.38868177e-01 4.47408036e-01 1.37041102e-01
 2.62256876e-01 5.68020076e-01 7.91845059e-01 1.19478918e-01
 2.76245965e-01 8.66414865e-01 5.19022729e-02 8.61404915e-01
 3.43439111e-01 8.44566694e-01 9.32945965e-01 1.07265080e-01
 9.09436886e-01 4.49294772e-02 9.37920549e-01 4.29301016e-02
 5.18871848e-01 1.93092191e-01 2.29645321e-01 9.56955708e-02
 3.17370090e-01]
[2.85531083e-01 7.01864534e-01 3.63800636e-01 2.90038319e-01
 9.78469018e-01 2.57301851e-01 8.13005295e-01 3.05045973e-01
 4.33714299e-01 8.02294004e-01 4.33525710e-01 3.59782875e-03
 3.85545493e-01 3.96768923e-01 6.81999047e-01 9.04271845e-02
 9.64427723e-01 4.49773114e-01 9.39912831e-01 6.37894203e-01
 9.11992347e-01 8.14096049e-01 4.33800594e-01 2.68757578e-02
 6.41474495e-01]
[9.17360273e-01 6.54973503e-01 3.92630597e-02 6.90763696e-01
 2.77043198e-01 3.37471188e-03 1.85650908e-02 5.95792747e-01
 5.49530684e-01 9.98706425e-01 2.44400332e-01 5.35412733e-01
 3.77815709e-02 8.95811274e-01 8.55204518e-01 6.69295055e-01
 7.06264556e-01 6.47527797e-01 4.51087145e-01 9.33843024e-01
 2.69679442e-01 9.24896363e-01 7.79325435e-01 7.53695959e-01
 1.97328952e-01]]
```

[4]:
```
# Defines a Pauli matrix and performs several operations on its and the transpose
sig_2 = numpy.mat([[0,-1j],[1j,0]])
# Matrix product of the Pauli matrix transpose with itself
prod = sig_2.T * sig_2
print("prod = \n",prod)
# Component-wise product
prod_elem = numpy.multiply(sig_2.T,sig_2)
print("prod_elemen = \n",prod_elem)
```

```
# Add third row of 1's and delete middle row of result
sig_2_mod = numpy.delete(numpy.vstack((sig_2,[1,1])),1,axis=0)
print("sig_2_mod = \n",sig_2_mod)
```

```
prod =
 [[-1.+0.j  0.+0.j]
 [ 0.+0.j -1.+0.j]]
prod_elemen =
 [[0.+0.j 1.-0.j]
 [1.-0.j 0.+0.j]]
sig_2_mod =
 [[ 0.+0.j -0.-1.j]
 [ 1.+0.j  1.+0.j]]
```

[5]:
```
# Define special matrix A as specified by the competency
A = numpy.diag([1,1,1,1],k=1) + numpy.diag([1,1,1,1],k=-1) + numpy.diag([math.
 ↪sqrt(i) for i in range(1,6)])
print("A = \n",A)
# Iterate over all entries and calculate sum by adding
A_sum = 0
for entry in numpy.nditer(A):
    A_sum += entry
print("sum of A = ",A_sum)
# Iterate over all columns and display sum using slicing and sum
for col_i in range(A.shape[1]):
    print("sum of column",col_i,"=",sum(A[0:A.shape[0],col_i]))
# Solve specified system of equations from the competency
b = numpy.ones((A.shape[0],1))
x = numpy.linalg.solve(A,b)
print("x = \n",x)
```

```
A =
 [[1.         1.         0.         0.         0.        ]
 [1.         1.41421356 1.         0.         0.        ]
 [0.         1.         1.73205081 1.         0.        ]
 [0.         0.         1.         2.         1.        ]
 [0.         0.         0.         1.         2.23606798]]
sum of A =  16.38233234744176
sum of column 0 = 2.0
sum of column 1 = 3.414213562373095
sum of column 2 = 3.732050807568877
sum of column 3 = 4.0
sum of column 4 = 3.23606797749979
x =
 [[-0.17237343]
 [ 1.17237343]
 [-0.48561297]
 [ 0.66873292]
```

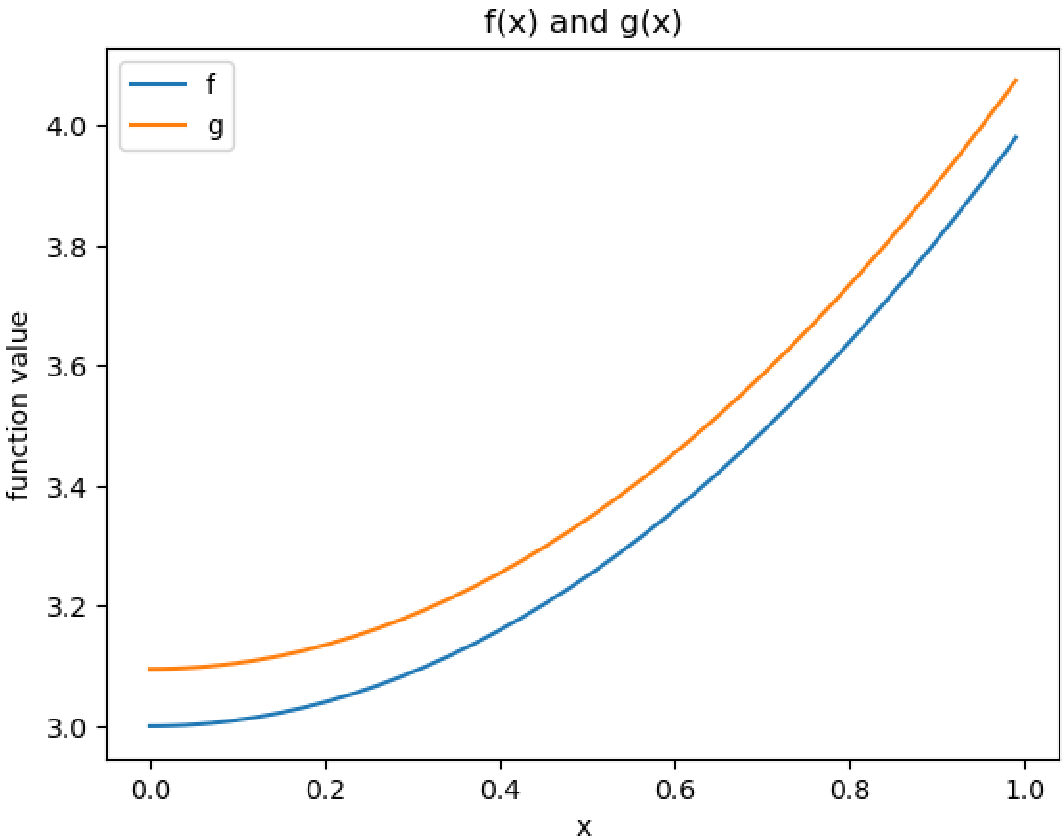
```
[ 0.14814714]]
```

[6]:
```
def f(x):
    return x*x + 3
```

[7]:
```
def g(x):
    return f(x) + numpy.random.rand()*0.1
```

[8]:
```
# Solves problem 5 of the competency
x_domain = numpy.arange(0,1,0.01)
f_range = f(x_domain)
g_range = g(x_domain)
# Declare new plot
plt.figure()
# Plot f and g off x_domain
plt.plot(x_domain,f_range)
plt.plot(x_domain,g_range)
# Labels and titles
plt.xlabel("x")
plt.ylabel("function value")
plt.legend(["f","g"])
plt.title("f(x) and g(x)")
```

[8]:
```
Text(0.5, 1.0, 'f(x) and g(x)')
```


f(x) and g(x)
f
g
function value
x
4.0
3.8
3.6
3.4
3.2
3.0
0.0
0.2
0.4
0.6
0.8
1.0

[ ]: