University of Canberra

Faculty of Science and Technology

**Programming for Data Science G (11521)**

**Week 5 Tutorial**

**List, Tuple and Assignment 1**

**Questions: Write Python code in Week5Tutorial.py for each question below**

* **Question 1**: Create a list of 100 elements like this [0, 1, 2, 3, 4, …, 99]

**Answer**

mylist = [i for i in range(100)]

print(mylist)

* **Question 2**: Create a tuple of 100 elements like this (0, 1, 2, 3, 4, …, 99)

**Answer**

mylist = [i for i in range(100)]

mytuple = tuple(mylist)

print(mytuple)

* **Question 3**: Change values of input\_list from string to number and output as output\_list
  + input\_list = ['2.1', '3.5', '4.8', '1.1', '2.0']
  + output\_list = [2.1, 3.5, 4.8, 1.1, 2.0]

**Answer**

input\_list = ['2.1', '3.5', '4.8', '1.1', '2.0']

output\_list = [float(x) for x in input\_list]

print(output\_list)

* **Question 4**: Change each element **x** in a list to **x / sum** where sum is total of all elements in that list. For example, mylist = [0, 2, 1, 3, 1, 2, 0, 1] and sum = 0+2+1+3+1+2+0+1 = 10 and mylist becomes [0.0, 0.2, 0.1, 0.2, 0.3, 0.2, 0.0, 0.1]

**Answer**

mylist = [0, 2, 1, 3, 1, 2, 0, 1]

sum = 0

for x in mylist:

sum += x

for i in range(len(mylist)):

mylist[i] /= sum

print(mylist)

* **Question 5**: Remove the first and last elements from a list. For example, my\_list = ['red', 0, 2, 1, 1, 2, 0, 1, 'blue'] becomes my\_list = [0, 2, 1, 1, 2, 0, 1]

**Answer**

my\_list = ['red', 0, 2, 1, 1, 2, 0, 1, 'blue']

my\_list = my\_list[1:-1]

print(my\_list)

* **Question 6**: Change 0 to 10 in [0, 1, 0, 2, 0, 1] to have [10, 1, 10, 2, 10, 1]

**Answer**

mylist = [0, 2, 1, 0, 1, 2, 0, 1]

mylist = [10 if x==0 else x for x in mylist]

print(mylist)

* **Question 7**: Combine list1 and list2 to have list3, list4 and list5 as follows
  + list1 = [2, 3, 1]
  + list2 = [4, 5, 2]
  + list3 = [2, 3, 1, 4, 5, 2]
  + list4 = [[2, 3, 1], [4, 5, 2]]
  + list5 = [(2, 3, 1), (4, 5, 2)]

**Answer**

list1 = [2, 3, 1]

list2 = [4, 5, 2]

list3 = list1.copy()

list3.extend(list2)

list4 = []

list4.append(list1)

list4.append(list2)

list5 = []

list5.append(tuple(list1))

list5.append(tuple(list2))

print(list1)

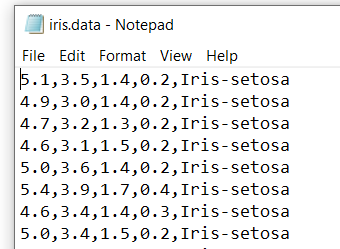
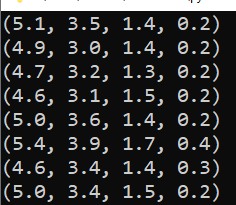
print(list2)

print(list3)

print(list4)

print(list5)

* **Question 8**:
  + Write a function (**read\_multi\_dim\_data**) in **io\_data\_module.py** that reads data from **iris.data** file and outputs a list of tuples where each tuple is a data sample.
  + Write a program in **Week5Tutorial.py** that calls the **read\_multi\_dim\_data** function to print out all 150 data samples. Below is example for the first 8 lines

For more information on the **iris** data set

* + Each line contains 4 real numbers and a string. The 4 numbers form a 4-dimensional data sample and the string is class label (the term ‘class’ here is for data classification, it’s not class in Python object-oriented programming).
  + There are 3 classes (Iris-setosa, Iris-versicolor, and Iris-virginica) in the Iris dataset.
  + There are 50 data samples in each class.
  + All 150 data samples are in 4-dimensional data space.

**Answer**

**io\_data\_module.py**

#Function to read multi-dimensional data from file and

#return data as a list of tuples

def read\_multi\_dim\_data\_file(filename):

dataset = [] #dataset is a python list

f = None

try:

f = open(filename, 'r')

while True:

line = f.readline()

if len(line) == 0: #end of file

break

line = line.replace('\n', '') #remove end of line \n character

string\_list = line.split(',') # coordinates in string format

if len(string\_list) < 2: #empty line

continue #skip it

string\_sublist = string\_list[:-1]

sublist = [float(x) for x in string\_sublist]

dataset.append(tuple(sublist))

except Exception as ex:

print(ex.args)

finally:

if f:

f.close()

return dataset

#end of function

**Week5Tutorial.py**

import io\_data\_module as iodata

data\_list = iodata.read\_multi\_dim\_data\_file('iris.data')

for sample in data\_list:

print(sample)

* **Question 9**: Write a Python program in **Week5Tutorial.py** that
  + reads all 150 data samples in **iris.data** and displays them on canvas using the function developed in Question 8 above, where **x** and **y** are the **first** and **second** values in each data sample, respectively, and
  + displays 3 centre samples centre\_1 = (5.1, 3.0, 1.1, 0.5), centre\_2 = (4.4, 3.2, 2.8, 0.2), and centre\_3 = (5.7, 3.9, 3.9, 0.8) on the same canvas with the **iris.data** data samples as seen below where red dots are iris data samples and black dots are centres.

**Answer**

**io\_data\_module.py**

#Function to show data on canvas

def show\_data(drawing\_list, xi=0, yi=1, colour='red', canvas=None, r=5, s=150, tx=300, ty=200):

for sample in drawing\_list:

x = sample[xi]

y = sample[yi]

x = x\*s + tx

y = y\*s + ty

if canvas != None:

canvas.create\_oval(x-r, y-r, x+r, y+r, outline = colour, fill=colour)

#end of function

**Week5Tutorial.py**

import io\_data\_module as iodata

import tkinter

#Open file and read data

data\_list = iodata.read\_multi\_dim\_data\_file('iris.data')

centre\_1 = (5.1, 3.0, 1.1, 0.5)

centre\_2 = (4.4, 3.2, 2.8, 0.2)

centre\_3 = (5.7, 3.9, 3.9, 0.8)

centre\_list = [centre\_1, centre\_2, centre\_3]

#Create canvas

top = tkinter.Tk()

C = tkinter.Canvas(top, bg="white", height=600, width=800)

#Display data

s = 150 #scale factor

r = 4 #radius

t1 = -500 #x translation

t2 = -200 #y translation

xi = 0

yi = 1

drawing\_list = data\_list.copy()

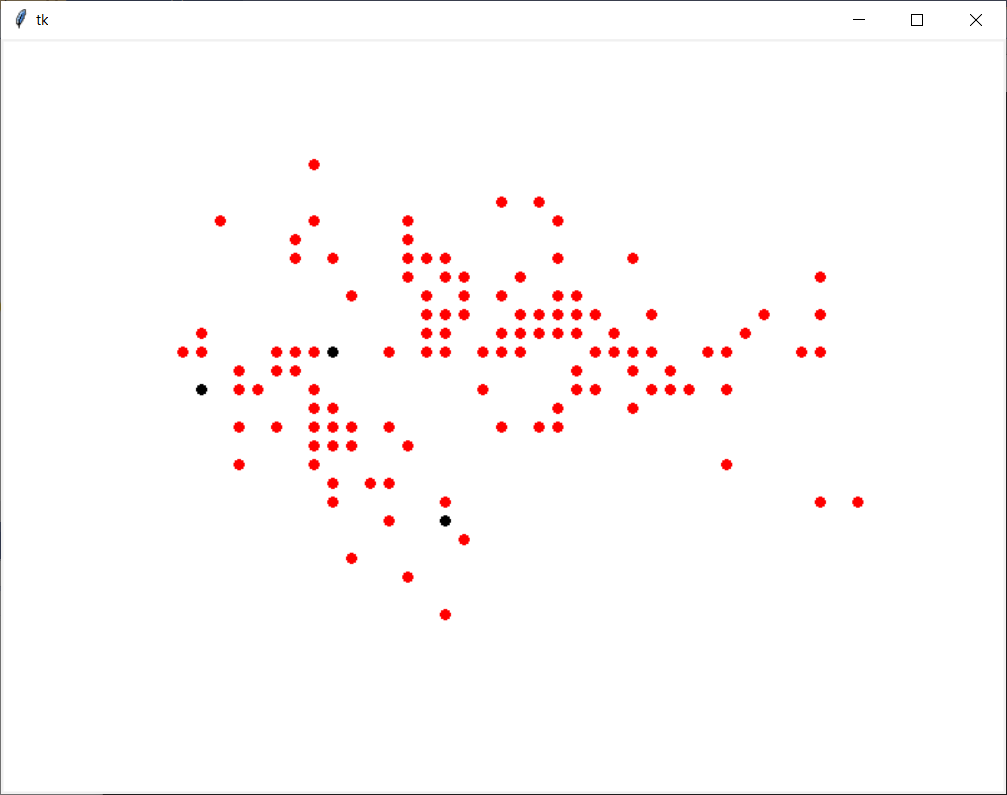
iodata.show\_data(drawing\_list, xi, yi, 'red', C, r, s, tx, ty)

drawing\_list = centre\_list

iodata.show\_data(drawing\_list, xi, yi, 'black', C, r, s, tx, ty)

C.pack()

top.mainloop()



**Total mark for assessment: 3%. Complete and submit all examples and questions: 2%, and lab attendance: 1%. Submit after the due date: -0.5% and -0.5% for each day after. The due date is end of this tutorial session.**