**Week 7 Tutorial**

**Objective: Familiarising Python programming techniques, including functions, file input/output, string manipulation, lists, tuples and dictionaries.**

1. **Calculate datastatistics.**

Write a program to calculate the sum, maximum and minimum values of a list. You need to define your own functions for calculating the above values instead of calling system functions. A skeleton of your program is provided below and you need to complete the empty functions

defmysum(x):

#Yourcodegoesfromheretocalculateandreturnthesumoflist x

defmymax(x):

#Yourcodegoesfromheretocalculateandreturnthemaximumvalue

defmymin(x):

#Yourcodegoesfromheretocalculateandreturntheminimumvalue

# x = [0,1,2,3,4,5,6,7,8,9]

x = range(10)

# expected result:sum= 45 max=9 min= 0

print"sum=",mysum(x)," max=",mymax(x)," min=",mymin(x)# x =[1,3,5,7,9]

x=range(1,10,2)

# expected result:sum=25 max=9 min=1

print"sum=",mysum(x)," max=",mymax(x)," min=",mymin(x)

# x = 100 random numbers, rand() is a function to generate random numbers from

# standard uniform distribution. Use help(rand) or ?rand or rand? to see help page on #randfunctioninCanopy.Rememberkeyinqandpressreturntoquithelppage.

x = rand(100)

# expected result: depending on the random values.

print"sum=",mysum(x)," max=",mymax(x)," min=",mymin(x)

# Vector norm, inner product anddistance.

## Write a program to calculate the so-called norm of vector, as well as the inner product and distance between two vectors. A vector is a mathematical term that can be represented as a list of numbers.Let

*x* [*x*1 ,*x*2,..., *xn* ] , *y* [*y*1 , *y*2,..., *yn* ] be two vectors of size n. The norm of vector *x* is written as || *x*||

## calculated by ||*x*||  .

*n*

∑

*i*1

*x*

2

*i*

*x* *x*... *x*

2

2

1 2

2

*n*

*n*

The inner product between vectors x and y is*x*,*y*calculated by *x*,*y*∑*xiyi**x*1*y*1*x*2*y*2...*xnyn*.

*i*1

## The distance between vectors x and y is given by *dist*(*x*,*y*) .

||*x*||2||*y*||2−2*x*,*y*

A skeleton program is provided below and you need to complete the empty functions. Square root of a

number *t* ,i.e. ,canbecalculatedbyusingmathlibrary'ssqrtfunction,i.e.sqrt(t))



*t*

defgetNorm(x):

#Yourcodegoesfromheretocalculateandreturnthenorm

def getInnerProd(x, y):

# Your code goes from here to calculate and return the inner product

def getDist(x, y):

# Your code goes from here to calculate and return the distance

# xVec= [0,1,2,3,4]

xVec =range(5)

# yVec= [4,3,2,1,0]

yVec=range(4,-1,-1)

# expected result: norm(x)= 5.4772 norm(y) =5.4772

print "norm(x)=",getNorm(xVec)," norm(y) =",getNorm(yVec) #expectedresult:innerprod(x,y)=10

print "innerprod(x,y) =", getInnerProd(xVec, yVec) # expected result: distance(x,y) = 6.3246

print "distance(x,y) =", getDist(xVec,yVec)

# Generate two random vectors of length 10 and calculate their distance. # Your code goes from here. Use rand() to generate randomvectors.

print"xvector=",x,"\nyvector=",y,"\n" print "distance(x,y) =", getDist(x,y)

# VoteCounting

## Write a function to count the votes from an input string. The votes are represented by 'Y's and 'N's in both upper and lower cases, and separated by ',' in between. Moreover, there might be extra spaces in the front or/and end of each vote. Your function should display a message for the voting results. The skeleton program is provided below and you need to complete function countVotes.

defcountVotes(votes):

#Yourcodegoesfromheretocounttheinputvotesandoutputvotingresult

votes = "N ,Y, Y,N,n, N , N , N ,n ,y, n,N,Y, y,Y,N, N , n ,y, N"

# the function call below should display the message # Reject: 20 votes in total, 7 accept and 13 reject countVotes(votes)

votes = "y,n,Y,y,y ,n, N, Y,N , N,y, n, Y,Y,y, N ,y, n, n , Y " #thefunctioncallbelowshoulddisplaythemessage

# Accept: 20 votes in total, 11 accept and 9 reject countVotes(votes)

# Wordcapitaliser

## Write a program to capitalise each word in a phrase except articles, coordinating conjunctions, prepositions and to in an infinity. To make things simpler, lowercase words in the list [*a, an, the, am, is, are, and, of, in, on, with, from, to*]. For example, for the input message "I am a good player", your program should return "I am a Good Player" as output. A skeleton program is provided in the following and you need to complete the empty function for word capitalisation. (Hint: you can create the list of words that must be lowercase as given above and decide if a word should be capitalised)

def capitalise(phrase):

#writeyourcodegoesfromheretoperformwordcapitalisation

# Output: "I am an Educator and a Researcher" capitalise("I am an educator and a researcher")

# Output: "Big Data is the Future of Information Technology" capitalise("big data is the future of information technology") # Output: "He Wants to Have Breakfast with Her in the Hotel" capitalise("Hewantstohavebreakfastwithherinthehotel")

# Top 10words

## For this exercise, you will complete a program that displays top 10 most frequently occurring words in an input file. We will use the file *data.txt*available to download from moodle. A skeleton program is provided in the following, where the sorting and displaying functionality has already been implemented.

You need to complete createDict() function which creates a dictionary of key value pairs contained in given input file as filename passed to the function. Each pair in the dictionary has a key, which is a word occurred in the input file, and a value, which is the number of occurrences of that word in the input file.

Your program should return this dictionary at output. Note that any return value of your function other than dictionary will cause the rest part of the program to fail.

# itemgetter is used by the sorted() system function

from operator import itemgetter

def createDict(filename):

# Your code goes from here to create and return a dictionary

myDict =createDict("bigdata.txt")

# sort the items of the dictionary by descending order of the second entry of each pai r (i.e.value in the key-valuepair)

# note that the return value is a list which is assigned to sortedListsortedList = sorted(myDict.items(), key=itemgetter(1), reverse = True) # print the top 10 entries in the sortedlist

forkey,valueinsortedList[:10]: print key, value