**Lesson 4**

Simple review of python

Basics:

* Lists
* Dictionaries
* Tuple
* Sets
* Strings
* Control Flow

Functions

Modules and Scoping Rules

Numeric Types

* Formats, Comparisons, Division, Precision
* Bitwise Operations

Function topics

* Function design
* Recursive functions

Small practices

1. Write a program which can compute the factorial of a given numbers.

Suppose the following input is supplied to the program:

8

Then, the output should be:

40320

Solution:

def fact(x):

if x == 0:

return 1

return x \* fact(x - 1)

x=int(raw\_input())

print fact(x)

2. Write a program which takes 2 digits, X,Y as input and generates a 2-dimensional array. The element value in the i-th row and j-th column of the array should be i\*j.

Note: i=0,1.., X-1; j=0,1,Y-1.

Example

Suppose the following inputs are given to the program:

3,5

Then, the output of the program should be:

[[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]]

Solution:

input\_str = raw\_input()

dimensions=[int(x) for x in input\_str.split(',')]

rowNum=dimensions[0]

colNum=dimensions[1]

multilist = [[0 for col in range(colNum)] for row in range(rowNum)]

for row in range(rowNum):

for col in range(colNum):

multilist[row][col]= row\*col

print multilist

3. Write a program that accepts a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically.

Suppose the following input is supplied to the program:

without,hello,bag,world

Then, the output should be:

bag,hello,without,world

Solution:

items=[x for x in raw\_input().split(',')]

items.sort()

print ','.join(items)

4. Write a program that accepts sequence of lines as input and prints the lines after making all characters in the sentence capitalized.

Suppose the following input is supplied to the program:

Hello world

Practice makes perfect

Then, the output should be:

HELLO WORLD

PRACTICE MAKES PERFECT

Solution:

lines = []

while True:

s = raw\_input()

if s:

lines.append(s.upper())

else:

break

for sentence in lines:

print sentence

5. Given a number, check if the number is a prime number by recursion

Input: n=11

Output: True

Input: n = 15

Output: False

Solution:

def isPrime(n, i=2):

if n <= 2:

return n==2

if n%i == 0:

return False

if i\*i > n:

return True

return isPrime(n, i+1)

6. Please write a binary search function which searches an item in a sorted list. The function should return the index of element to be searched in the list.

Solution:

def bin\_search(li, element):

bottom = 0

top = len(li)-1

index = -1

while top>=bottom and index==-1:

mid = (top+bottom)/2

if li[mid]==element:

index = mid

elif li[mid]>element:

top = mid-1

else:

bottom = mid+1

return index

li=[2,5,7,9,11,17,222]

print bin\_search(li,11)

print bin\_search(li,12)