Python HW #6

Name: Chris Welch

1. Change the order of an input string based on Unicode equivalent.

Write a function Reorder that will take an input string of a maximum length of 20 characters and create a new string that reorders the string based on it's binary Unicode character equivalent, e.g. 'a' = 0x61, 'A' = 0x41, from smallest value to largest value. In this example, 'A' would come before 'a', as 0x41 is a smaller number than ox61. Note, use python logical operators to do the comparison. Do not try to create a Unicode lookup table. For your code, use the string "sum=(x*259)/average" as your input string to your function.

```python
#-----Problem #1--------------#
## A-Z = 65-90     # a-z = 97-122
def Reorder():
    stringGetter = input("Enter a String, 20 Characters or less:\t\t")
    while len(stringGetter) > 20:
        stringGetter = input("20 Characters or less! Enter a String:\t\t")
    stringBuffer= []
    orderedString = ''
    for x in stringGetter:
        stringBuffer.append(str(ord(x)))
    buffList = []
    for x in stringBuffer:
        buffList.append(int(x))
    stringBuffer.sort()
    buffList.sort()
    print(buffList)
    for x in buffList:
        orderedString += (str(' 0x') + str(x))
    print(orderedString)

Reorder()
```

2. Create two lists from a dictionary

Write a function Delist that takes an input dictionary and creates a new list based on the dictionary keys only and a new list based on the dictionary values only. Return the two lists created.

For example

{ “A”: 0, “B”: 1, “C”:2, “D”:3}

returns the lists

[“A”, “B”, “B”, “D”]
[0, 1, 2, 3]

```python
#-----Problem #2--------------#
newdict = { 'A': 0, 'B': 1, 'C':2, 'D':3}
listKeys = []
listVals = []

def Delist(newdict, nListKeys, nListVals):
    for x in newdict.keys():
        nListKeys.append(x)
    for x in newdict.values():
        nListVals.append(x)
    listKeys.sort()
    listVals.sort()
    return nListKeys, nListVals

Delist(newdict, listKeys, listVals)
print(listKeys)
print(listVals)
```

3. 99 Bottles of Beer on the Wall song, recursively

Write a function Recurse that prints out the verses of the famous song, Lyrics of the song 99 Bottles of Beer (shown below). You must called this function recursively.

99 bottles of beer on the wall, 99 bottles of beer.
Take one down and pass it around, 98 bottles of beer on the wall.

98 bottles of beer on the wall, 98 bottles of beer.
Take one down and pass it around, 97 bottles of beer on the wall.

97 bottles of beer on the wall, 97 bottles of beer.
Take one down and pass it around, 96 bottles of beer on the wall.

.
.
.

2 bottles of beer on the wall, 2 bottles of beer.
Take one down and pass it around, 1 bottle of beer on the wall.

1 bottle of beer on the wall, 1 bottle of beer.
Take one down and pass it around, no more bottles of beer on the wall.

No more bottles of beer on the wall, no more bottles of beer.
Go to the store and buy some more, 99 bottles of beer on the wall.

```
#-----Problem #3--------------#
def Recurse(beerNum):
    if beerNum == 1:
        print("1 bottle of beer on the wall, 1 bottle of beer.")
        print("Take one down and pass it around, no more bottles of beer on 
the wall.\n")
        print("No more bottles of beer on the wall, no more bottles of beer.")
        print("Go to the store and buy some more, 99 bottles of beer on the 
wall.\n\n")
    else:
        print("%d bottles of beer on the wall, %d bottles of beer." % 
(beerNum, beerNum))
        print("Take one down and pass it around, %d bottles of beer on the 
wall.\n" % (beerNum - 1))
        beerNum -= 1
        Recurse(beerNum)

beers = 99
Recurse(beers)
```