Richard Hayes Crowley

06/17/2021

CSC\_242\_Lab\_04

**SOURCE CODE:**

*# Richard Hayes Crowley*

*# cryptography ( shuffle elements in a list )*

*import* random

*# for random shuffle*

strMsg = "The Courier is En Route with the Documents"

*# the plaintext secret message*

print("~\*~\*~\*~\*~\*~\*")

print(f"Plaintext message:\n{strMsg}")

*# convert the plaintext to lower case*

strMsg = strMsg.lower()

print("~\*~\*~\*~\*~\*~\*")

print(f"Plaintext message in lowercase:\n{strMsg}")

*# replace any spaces in the message with an "x"*

strMsg = "".join(["x" *if* char == ' ' *else* char *for* char *in* strMsg])

print("~\*~\*~\*~\*~\*~\*")

print(

f"Plaintext msg with spaces replace with 'x':\n{strMsg} ")

*# populate the list with the characters comprising the message*

plainTxtList = [char *for* char *in* strMsg]

print("~\*~\*~\*~\*~\*~\*")

print(f"Populate the list:\n{plainTxtList}")

*# declare the cipher list, create new ref*

scrambledList = list(plainTxtList)

*# shuffle it*

random.shuffle(scrambledList)

print("~\*~\*~\*~\*~\*~\*")

print(f"Perform a Random shuffle:\n{scrambledList}")

*# determine the length of the list*

print(f"Length of the scrambled list:\n{len(scrambledList)}")

*# print in blocks of seven letters per row, playing around a bit here with nested lists... more straightforward ways to do this, I'm sure.*

cipherList = [[]]

print("~\*~\*~\*~\*~\*~\*")

print("Ciphertext:")

*for* index *in* range(len(scrambledList)):

*if* index != 0 and index % 7 == 0:

cipherList.append([scrambledList[index]])

*else*:

cipherList[index //

7].append(scrambledList[index])

cipherListStr = "\n".join(["".join(item) *for* item *in* cipherList])

print(cipherListStr)

*# performing decryption... I could create a cipher that's position based and return it as an object, which would be useful if we wanted to modularize this program, however for purposes of this assignment I'll just decrypt in situ*

decryptedList = []

*for* idx, char *in* enumerate(scrambledList):

*if* plainTxtList[idx] == "x":

decryptedList.append(" ")

*else*:

decryptedList.append(plainTxtList[idx])

print(f"Performing decryption...")

print("".join(decryptedList).capitalize())

**… output on next page**

**OUTPUT  
  
Text

Description automatically generated**