Weekly Exercise 5 – Encryption-Decryption

• **Problem statement:**
  • Write your own encryption/decryption program

• **Requirements:**
  • You must create your own encryption/decryption algorithm (you cannot use the ones seen in class or anywhere else – textbooks, Internet, etc…)
  • You must encrypt the user’s message so that one cannot guess the plaintext by looking at the ciphertext
    • So, a solution like reversing the characters in the message won’t do!
  • You must use the **starter program** found on the next slide and you cannot modify its structure
  • You must create a function `encrypt()` and a function `decrypt()`, and implement them in the starter program
  • Note that the starter program asks the user to enter a message (plaintext) to encrypt/decrypt until the user wants to stop
  • All these requirements will be demonstrated in class this week in Lectures 12 and 13
# Encryption/Decryption
# Description: Encrypt plaintext and decrypt ciphertext using my own algorithm
# Author:
# Date:

# function encrypt()
def encrypt(plain):
    # function decrypt()
def decrypt(cipher):

# Main part of program
notFinished = True;
while notFinished:
    plainText = input("Please, enter a message to encrypt or S to stop: ")
    if plainText != 'S' and plainText != 's':
        # Call the encrypt function with plainText
        # The encrypt function returns ciphertext
        print("\tLet's encrypt your message ... ")
        cipherText = encrypt(plainText)
        
        # Print the ciphertext
        print("\t'{}' becomes '{}'.".format(plainText, cipherText))
        
        # Call the decrypt function with ciphertext
        # The decrypt function returns plaintext
        print("\tLet's decrypt your encrypted message ... ")
        plainText = decrypt(cipherText)
        
        # Print the ciphertext and plaintext
        print("\t'{}' becomes '{}'.".format(cipherText, plainText))
    else:
        notFinished = False;

print("Bye!")
How to proceed:

• Follow the process demonstrated in class:
  • Use Repl.it Python3
  • In the text editor
    • Write a complete header
    • Write an algorithm in English as Python comments
    • Translate your comments into Python 3 statements
    • Make sure your program solves the problem stated in the **Problem Statement** and satisfies the **Requirements**
  • Press the **Run** button to execute your program. Enter as many different responses to test as much of your code as possible

• Press the **Shared** button to get the link

• Submission: Copy and paste the link in **CourSys**

• Due this Thursday, June 6 at 3pm

• If you have any issues or questions, post them on Piazza or ask a TA during Open Lab hours or the instructor during her office hours

• You must work on your own.

• **No late submission** will be accepted!

• Make sure you test your link – if it does not work, the TA cannot mark your work and you get 0. 😞
A Note about CourSys

- You can submit your work as often as you wish on CourSys, i.e., you can make many submissions.
- The last submission is the one that is marked – make sure your last submission is not late!

- Your grade and comments can be found on CourSys:

Weekly Exercise 3

<table>
<thead>
<tr>
<th>Status</th>
<th>grades released</th>
</tr>
</thead>
<tbody>
<tr>
<td>Due date</td>
<td>Thu May 23 2019, 15:00</td>
</tr>
<tr>
<td>Percentage</td>
<td>1.00%</td>
</tr>
<tr>
<td>Grade Status</td>
<td>graded</td>
</tr>
<tr>
<td>Your Grade</td>
<td>1.00/1.00 (100.00%) [Additional Marking Details]</td>
</tr>
</tbody>
</table>

Click here for comments!