COSC2429 Intro to Programming

Lab 11: Putting Things Together

Objective: Apply all the Python programming concepts learned so far (loop, selections, strings, list, files, dictionaries, etc.) to solve problems.

1. Write a program, which will find all such numbers between 1000 and 3000 (both included) such that each digit of the number is an even number. The numbers obtained should be printed in a comma-separated sequence on a single line.

2. Write a program that computes the factorial of a given integer n knowing that:

   n! = n * (n – 1) * (n – 2) … * 2 * 1

   Suppose the following input is supplied to the program:
   Enter an integer: 8
   Then the output should be:
   8! = 40320

3. Write a function that implements a substitution cipher. In a substitution cipher one letter is substituted for another to garble the message. For example A -> Q, B -> T, C -> G etc. The function accepts two parameters: the message you want to encrypt and a string that represents the mapping of the 26 letters in the alphabet, e.g. “QTGABCDEFHIJKLMNOPRSUVXYZ”. The function returns a string that is the encrypted version of the message.

4. Write a function that decrypts the message from the previous exercise. It should also accept two parameters: the encrypted message, and a string representing the mapping for decryption. The function returns a string that is the original unencrypted message.

5. Write a program that accepts a sentence and calculate the number of letters, digits and punctuation characters.
   Suppose the following input is supplied to the program:
   Enter a sentence: hello world! 123&
   Then, the output should be:
   LETTERS 10
   DIGITS 3
   PUNCTUATIONS 2

6. A website requires the users to input username and password to register. Write a program to check the validity of password input by users. Following are the criteria for checking the password:
   a. Minimum length of transaction password: 6
   b. Maximum length of transaction password: 12
   c. At least 1 letter between [a-z]
   d. At least 1 letter between [A-Z]
   e. At least 1 number between [0-9]
   f. At least 1 character from [!@#$%^&*]

   Your program should accept a sequence of comma separated passwords and will check them according to the above criteria. Passwords that match the criteria are to be printed, each separated by a comma.

   Example:

   If the following passwords are given as input to the program:

   Enter passwords: ABd1234@1,a F1#,2w3E*,2We3345

   Then, the output of the program should be:

   Valid passwords: ABd1234@1