# CSC1015F Assignment 8: Recursion

## Assignment Instructions

This assignment involves constructing Python functions that use recursion. You must not use loop constructs (such as 'for' and 'while') in your solutions.

## Question 1 [25 marks]

A palindrome is a word, phrase, number, or other sequence of symbols or elements, whose meaning may be interpreted the same way in either forward or reverse direction (see Wikipedia).

Write a Python program called 'palindrome.py' that uses a recursive function to calculate whether or not a string is a palindrome (reads the same if reversed).

You may NOT use iteration, string slicing or any other technique to reverse the string without using recursion!

Sample I/O:

```
Enter a string:
able was I ere I saw elba
Palindrome!
```

Sample I/O:

```
Enter a string:
elba is a noob
Not a palindrome!
```

## Question 2 [25 marks]

Write a program called 'pairs.py' that uses a recursive function to count the number of pairs of repeated characters in a string. Pairs of characters cannot overlap.

*You MUST NOT use any form of loop in your program! (We may have mentioned this.)*

Sample I/O:

```
Enter a message:
hello, Salaama
Number of pairs: 2
```

## Question 3 [25 marks]

Write a program called 'encrypt.py' that uses a recursive function to encrypt a message by converting all lowercase characters to the next character (with z transformed to a).

*You MUST NOT use any form of loop in your program! (Yep, we've definitely mentioned this.)*

Sample I/O:

```
Enter a message:
hello world
Encrypted message:
ifmmp xpsme
```

## Question 4 [25 marks]

Write a program called 'palindromeprimes.py' that uses recursive functions to find all palindromic primes between two integers *N*, *M*, supplied as input. (start and end points are included).

- A palindrome number is a number that reads the same from the front and the back. Examples are: 212, 44, 9009, 4567654.
- To calculate whether a number is a palindrome or not, incorporate your answer to question 1.
- A prime number is a number greater than 1 that is only divisible by 1 and itself. Examples are: 3, 11, 313.

Some examples of palindromic primes are: 11, 191, 313.

You may assume it's always the case that *N>1*, and that *N≤M*.

*You MUST NOT use any form of loop in your program! (Yes, we're pretty obsessive about this.)*

Add the following lines at the top of your program to increase the amount of recursion that Python will allow:

```
import sys
sys.setrecursionlimit (30000)
```


Sample I/O:

```
Enter the starting point N:
200
Enter the ending point M:
800
The palindromic primes are:
313
353
373
383
727
757
787
797
```


## Submission

Create and submit a Zip file called 'ABCXYZ123.zip' (where ABCXYZ123 is YOUR student number) containing palindrome.py, pairs.py, encrypt.py, and palindromeprime.py.


# END