# Homework 4 Regular expressions

*NB: You can edit the first two questions to match your own school’s tutor group naming conventions if required.*

1. Julia has written some Python code, but it won’t run. She has made 3 syntax errors – highlight them and explain each one to Julia. [3]

import regEx  
tutorGroup = input(“Enter your tutor group – e.g. 10MC: ”)  
valid = re.match([0-9][0-9][A-Z][a-Z], “tutorGroup”)  
if valid:  
 print(“OK!”)  
else:  
 print(“Invalid!”)

1.

2.

3.

1. George wants to alter Julia’s program so that it will accept tutor groups with 1 or more numbers (e.g. 7MC, 10MC, 12MC). Write just the one line of code that George needs to edit and explain what he needs to do. [2]

1. Darcey has been looking at a program that will collect and then validate IP addresses   
   (e.g. 192.168.0.37). IP addresses are always made up of 4 block of digits, separated by full stops. Add comments to explain how the program works.  
   NB: The program is only meant to check for the correct pattern, it doesn’t check if each element is between 0 and 255) [5]  
     
   #   
   #   
   import re  
   ip = input(“Enter IP address: ”)  
   #   
   #   
   #   
   valid = re.match(“[0-9]+\.[0-9]+\.[0-9]+\.[0-9]”,ip)  
   #   
   #   
   if valid:  
    print(“IP address OK”)  
   else:  
    print(“IP address invalid”)

4. There are four valid telephone number formats in the UK. These are:

(01nnn) nnnnnn

(01n1) nnn nnnn

(011n) nnn nnnn

(02n) nnnn nnnn

Write regular expressions for each of these formats. The first one is done for you. Note that [0-9]{6} means “exactly 6 instances of any digit between 0 and 9”. [3]

pattern1 = “(01[0-9]{3}) [0-9]{6}”

pattern2 =

pattern3 =

pattern4 =

Write a Python statement to test whether a phone number conforms to one of the patterns. If it does, print “valid phone number”. If it does not, print “Invalid phone number”. (Tip: You could use an IF..ELIF” statement) [5]

[Total 18 Marks]