CAB431 Workshop (Week 2)
Basic I/O operations and Pre-processing Textual Data

Objectives:

- Getting familiar with Python for doing practical questions.
- Understand Python Basic (see week 1 session 1.5 Activity 2 "Read Python Basic document"):
    - Python commands and statements
    - Data structures: list and dictionary
    - Functions and Classes
    - Reading and writing files
- Understand how to use Python to do basic data pre-processing for text documents.
- Please note that this unit does not teach programming principles (the assumed knowledge or prerequisites: CAB201), but nothing beyond the capabilities of someone who has taken some basic computer science and programming classes. Please contact the unit coordinator if you have any difficulties for using Python.

Task 1: Read week1 reading "Python_basic.pdf" and test all programs used in this file if you haven't done it in week 1 and discuss any problem with your tutor, use the Slack or send emails to the unit coordinator: y2.li@qut.edu.au

Task 2: Write a program that loads (read) an XML document (741299newsML.xml) and prints out the **itemid** and the number of words in <**text**> of the document.

The following is the Format of the Document:

<newsitem xml:lang="en" date="1997-07-20" id="root" **itemid="741299"**>
<title>BELGIUM: MOTOR RACING-LEHTO AND SOPER HOLD ON FOR GT VICTORY.</title>
<headline>MOTOR RACING-LEHTO AND SOPER HOLD ON FOR GT VICTORY.</headline>
<dateline>SPA FRANCORCHAMPS, Belgium</dateline>
<text>
J.J. Lehto of Finland and Steve Soper of Britain drove their ailing McLaren to victory in the fifth round of the world GT championship on Sunday, beating the Mercedes of German Bernd Schneider and Austrian Alexander Wurz by 15 seconds.

Their victory enabled them to open up a 16-point lead in the overall standings over Schneider, who mounted a strong challenge on the struggling leaders in the final minutes of the four-hour race.
But Soper, struggling with the car's handling caused by a broken undertray, just managed to hold on for the win.
Lehto had opened up a lead of over 90 seconds during a mid-race downpour in the Ardennes mountains.
"I thought that everyone else was driving on dry-weather tyres," he joked afterwards.
"We swapped to rain tyres at exactly the right time and I was able to push hard and open up a big lead."
Third to finish was the Porsche of France's Bob Wollek and Yannick Dalmas and Belgian Thierry Boutsen.
The Belgian, a former Formula One driver, switched from the car he normally shares with German Hans Stuck following a power-steering failure on his own car.
</text>
<copyright>(c) Reuters Limited 1997</copyright>
</newsitem>

Task 3 (Optional): Design a python solution for generating a consensus string.

A **consensus string** x is a string of length n formed from a collection by taking the most common symbol at each position; the j^{th} symbol of x therefore corresponds to the symbol having the maximum value in the j^{th} column of the profile matrix. Of course, there may be more than one most common symbol, leading to multiple possible consensus strings.

For example,

		A	T	C	C	A	G	C	T
		G	G	G	C	A	A	C	T
		A	T	G	G	A	T	C	T
DNA Strings		A	A	G	C	A	A	C	C
		T	T	G	G	A	A	C	T
		A	T	G	C	C	A	T	T
		A	T	G	G	C	A	C	T
	A	5	1	0	0	5	5	0	0
Profile	C	0	0	1	4	2	0	6	1
	G	1	1	6	3	0	1	0	0
	T	1	5	0	0	0	1	1	6
Consensus		A	T	G	C	A	A	C	T

Given: A collection of DNA strings of equal length (see the attached file seq1.tex and seq3.tex).

Return: A consensus string and profile matrix for the collection. (If several possible consensus strings exist, then you may return any one of them.)

Sample file:

```
>Rosalind_1
ATCCAGCT
>Rosalind_2
GGGCAACT
>Rosalind_3
ATGGATCT
>Rosalind_4
AAGCAACC
>Rosalind_5
TTGGAACT
>Rosalind_6
ATGCCATT
>Rosalind_7
ATGGCACT
```

Sample Output

```
ATGCAACT
A: 5 1 0 0 5 5 0 0
C: 0 0 1 4 2 0 6 1
G: 1 1 6 3 0 1 0 0
T: 1 5 0 0 0 1 1 6
```