INFO-6079 W2023   Security Applications w/Python
FINAL PRACTICAL TEST

```
DATE      WINTER 2023
DUE       April 12, 2023
```

Aids Permitted:
- Laptop
- all code run in class, labs and assignments
- all class PowerPoint Presentations
- Peer to Peer communication is not permitted
- You may not use the network or web except to upload your solutions at the end of the test

Marks: 40

Weighting 15%

Instructions:
- READ the complete test before starting.
- Get an idea of what to do!!!
- Answer the questions by creating Python or text files as directed
- Make sure your name and Student ID are in all submitted files
- Python files *MUST BE ABLE TO BE RUN* on any PC
  - *Do Not Use your DESKTOP as a reference*
- Zipped files will NOT be accepted .txt or .py file only
- upload files (.txt and .py) to FINAL PRACTICAL file naming
  eg. (bhansen)_final_test.txt your FOLID text file
      (bhansen)_final_text.py "   python file.

Part 1

	Answer the questions - Enter answers into a text file Once complete – upload the text file	Marks
1.	Indicate your name, student ID	2

2.	Explain the use of Sockets
	   1. why do we need a Client and Server?
	   2. why are IP Addresses important?
	   3. how are errors handled?
	   4. why is the SERVER always listening?
		2
		 2
		 2
		 2

	Total	10

Part 2

	Answer the questions - Create the code in a .py file Once complete – upload the Python file	Marks
1.	Create a log file	2
2.	Insert name and Student ID	2
3.	Insert programmed Current Date	2
4.	Add code to Read data from a text file and insert the data into the log file See Week 7 Lab *ipaddress_shuffled_list.txt*	4
	TOTAL	10

School of Information Technology
Fanshawe College
INFO-6079 Final Practical Test

FANSHAWE
School of Information
Technology

Part 3

	Answer the questions - <mark>Create the code in a .py file</mark> Once complete - upload the Python file	Marks
1.	Using the same Python file as created in PART 2	2
2.	Have a class private variable.	1
3.	Create Constructor __init__ () that saves the variables to private instant variable.	2
4.	Create a CLASS to be used to add numbers	1

5.	Create ADD_NUMBER()
	  1. Private function
	  2. Takes a variable (from USER INPUT)
	  3. adds the variable - until an empty
	     variable is entered (the user just
	     pressed ENTER)
	  4. display the total back to the user
		4

6.	Import modules
	  1. import os
	  2. import platform
		1

7.	Print Current Working Directory (CWD)	1
8.	Print file(s) and directories in the current Directory	2
9.	Retrieve the Laptop OS information and write the information to the log file - created in step 2	2
10.	Retrieve current version of Python and write the information to the log file - created in step 2	2
11.	Create a new Folder and write the name of the Folder to the log file - created in step 2	2
	TOTAL	20