Task 21-01

- Create a Jupyter Lab Notebook called **octal_converter.ipynb**
- In cell 1, convert the octal number 2323 to base 10 (decimal) by looping, and display both numbers
- In cell 2, convert the decimal number you calculated in cell 1 back to octal by looping, and display both numbers
- In cell 3, convert the octal number 2323 to base 10 (decimal) without looping, and display both numbers
- In cell 4, convert the decimal number you calculated in cell 3 back to octal without looping, and display both numbers
- Upload your solution to the BNL QIS101 SharePoint site

Task 21-02

- Using the Digital App, create a circuit file called **full_adder.dig**
  - Using only 1 **OR**, 2 **AND**, 2 **XOR** gates, plus five I/O pins, design a **FULL ADDER** circuit
  - The circuit has **3 input lines**, and **2 output lines** to indicate the sum and if there needs to be a carry to the next column
  - Verify your circuit by having Digital create a truth table
- Upload to the BNL QIS101 SharePoint site your circuit (.dig) file

FULL ADDER Truth Table

INPUT				OUTPUT	
A	B	C_{in}		S	C_{out}
0	0	0		0	0
0	0	1		1	0
0	1	0		1	0
0	1	1		0	1
1	0	0		1	0
1	0	1		0	1
1	1	0		0	1
1	1	1		1	1

Task 21-03

- Create a Jupyter Lab Notebook called **majority_vote.ipynb** to simulate the circuit and generate the truth table of the majority vote circuit shown on slide #24
- In Cell 1, declare all the necessary arrays for the Boolean states and logic gates required to simulate all permutations of input and output states
- In Cell 2, implement the complete classical digital logic circuit using Numpy matrix algebra functions
- In Cell 3, generate and display the full truth table for the circuit
- Upload your solution to the BNL QIS101 SharePoint site