LAB #3
Binary, Hex, and Octal Conversions

1. **Pre-Lab/Groupwork**: Thanks to a TA David Snelling, here is a helpful guide for understanding this material: [Exam1Review1.pdf](#)

   Sticking with paired programming in our labs, get into a group of 2 (maybe try to meet/pair with someone different), and discuss how you would convert these decimal numbers into binary (base 2), hexadecimal (base 16), and octal (base 8) numbers. Write your solutions on a piece of paper, and show your work!!!

   17, 29, 33, 46, 54, 96, 100, 255, 256, 1023, 1024, 1028, 4096

   Convert the following binary numbers into decimal numbers, and show your work!!!

   11, 101, 111, 1011, 1101, 1111, 10110110, 11111111, 10010010, 11100011

   Now, let’s practice calculating ranges for different data types that represented by a specified number of bytes/bits.

   - A character: 1 byte = 8 bits
   - A short integer: 2 bytes
   - An integer: 4 bytes
   - A long integer: 8 bytes

   Determine the range of signed numbers, i.e. positive and negative, and unsigned numbers, i.e. only positive, for each data type listed above.

   **NOTE: Make sure that both members in your group put their name on the piece of paper to get full credit for being in the lab!**

2. **Testing your solution**: Write Python code to test your handwritten solutions.

   Start a new Python project in Visual Studio. If you have not been able to get Visual Studio to work but have downloaded Python, then use the built-in Python IDLE. (Go to Python 3.2-> IDLE (Python GUI)).

   Write a python program that assigns the above decimal numbers to a variable, and then prints the binary, hexadecimal, and octal numbers of the variable.

   **Example:**
   ```python
   num = 17
   print("17 as bin, hex, oct:")
   print(bin(num))
   print(hex(num))
   print(oct(num))
   ```
Now, write a python program that assigns the above binary numbers to variable, and then prints the decimal value of the binary number.

**Example:**
num = 0b11
print("binary 11 in decimal is:")
print(num)

**Compare your handwritten solutions with your Python solutions** to see if you got the correct answer. For those answers where your handwritten solution differed from the Python solution, document what caused this error. For example, did your make a typo when entering the number in Python or a logical error when calculating your solution by hand?

3. In order to receive credit for your lab, show your Python code to a lab TA before leaving the lab, **otherwise you’ll receive zero points!!!**