CSI 250 - Homework Set #4

To receive credit on this assignment, you need to show your work. If you believe there is "no work to show" then explain how you derive your answer instead. Spacing in binary digits is to help with parsing the long strings of numbers.

1. Convert $(27)_{10}$ into 8-bits in the following formats:
   (a) unsigned binary
   (b) hexadecimal
   (c) excess-127
   (d) two's complement
2. Convert $(-27)_{10}$ into 8-bits in the following formats:
   (a) excess-127
   (b) two's complement
3. Convert $(1000\ \ 0000)_2$ from the following formats into the specified formats:
   (a) Convert from unsigned binary to hexadecimal
   (b) unsigned binary to decimal
   (c) excess-127 to decimal
   (d) two's complement to decimal
4. Show $(-1)_{10}$ in the following formats:
   (a) 2 bit two's complement
   (b) 4 bit two's complement
   (c) 8 bit two's complement
5. Convert $(2020.09375)_{10}$ into binary.
6. Convert $(0011\ \ 0110.1101\ \ 1000)_2$ into decimal.

7. The overflow flag is set in a processor when two's complement signed integers give an answer that cannot possibly be correct. For example, overflow in addition might occur if the result of two negatives added together yields a positive or two positives added together give a negative. In this question consider 32 bit two's complement integers. You have access to the most significant bit (the bit worth 2^{31} as A_{31} of one addend and B_{31} as the other addend. R_{31} is the most significant bit of the result. The least significant bits (the 'ones' place worth $2^0 = 1$) are accessed as A_0, B_0, and R_0. You may access other elements as needed by using the proper subscripts (there are 96 total bits here). Make a Karnaugh Map (Hint: 96 bits are available but our K-Maps max out at 4) and create a simple equation to compute the overflow flag/bit for 32 bit integer addition. Overflow is 1 when the calculation is "wrong." Overflow is 0 otherwise.