CSCI 2330 – Floating Point Exercises

1. Using an 8-bit IEEE floating point representation (with k=4 exponent bits and 3 fractional bits), convert **00110100** into a decimal value.

2. Using the same 8-bit representation, convert **10000101** into a decimal value (working with a fraction here is advisable).

3. If **d** is a double, does (d < 0.0) imply that ((d * 2) < 0.0)? Remember that this property is <u>not</u> guaranteed for ints.

4. Excluding infinity, what is the decimal value of the largest 32-bit IEEE floating point number? You should be able to write down an exact expression (unsimplified is fine).

5. IEEE 754 encodes the exponent value **E** using the **exp** bits as an unsigned value from which **bias** is subtracted. An alternative approach would be to just make the **exp** bits encode a signed value and get rid of the **bias** term. Is there a reason to prefer the **unsigned - bias** approach?

Hint: consider smaller and larger floating point values encoded in this format (versus the alternative) and think about the ordering of their raw bit patterns (e.g., the three bit pattern 001 comes "before" 010, etc.)