Nate Flasher

ICS

Dr. Nurkkala

Midterm Remediation

Problem 5)

**Answer 1**: This was already correct.

**Answer 2**: The reason I arrived at the answer, 1111111000000001 is simply because I got the special values for floating-point binary representations of NaN (Not a Number) and infinity swapped in my head. The current representation of infinity of floating-point values in binary is all 1’s for the exponent bits and all 0’s for the fractional bits. Therefore, my answer to this problem is 1111111000000000 as this represents infinity rather than NaN.

**Answer 3**: For this problem my answer was 0100111010100110. My main mistake was how I calculated the value of exp. I calculated the value of E to be 8 because I included the bits after the decimal for the binary value 101010.**011** (42.375). Those bits I underlined and bolded I realize I should not have taken into account. Because I counted those, I got the E value to be 8 instead of 5. I added 8 to the bias of 31 and got an exp value of 39. That translated to 100111 in binary, which put into the final encoding was 0100111010100110 which was wrong. Instead, the value of E should have been 5. Then adding that to the bias, exp becomes 36. 36 is 100100 in binary. Finally, we have the 0 for the signed bit, 100100, for the exp bits, and 010100110 for the fractional bits. Putting that altogether my new correct solution is: 0100100010100110.

Problem 6)

For this problem I misunderstood that I that I needed to use the equation/formula: *Mem[Reg[Rb] + (S × Reg[Ri ]) + D]* to calculate the address in hex where the mov instruction will read from, and instead thought one of the registers listed were the options to choose from. Therefore, to correctly solve this problem, I did 0x8 + 0x1000 + (0x4 \* 0x3) = 0x1014.