

## CPTS 260 Intro to Computer Architecture, Quiz 2 Solutions

**Q 1** What is the minimum number of bits required to represent 100 in 2's complement binary?

Step	Q	R
100/2	50	0
50/2	25	0
25/2	12	1
12/2	6	0
6/2	3	0
3/2	1	1
1/2	0	1

The binary representation is **1100100**. However, we have to represent it in the 2's complement representation, which means that the MSB (leftmost bit) shows the signs of the number. As a result, we need to prefix a 0 to result we obtained. The 2's complement representation is **01100100**, which takes 8 bits.

**Q 2** Double precision floating point representation completely eliminates overflow and underflow present in single precision floating point representation.

**Solution:** This is false because double precision floating point can have overflow or underflow as well. Double precision floating points can represent a larger range of number when compared to single precision, however, this does not eliminate overflow/underflow.

**Q 3** Consider a integer binary multiplication with a  $m$  bit multiplier and  $n$  bit multiplicand. What is the maximum number of bits required to store the results of the multiplication?

**Solution:** The number of bits required is  $m + n$ .

**Q 4** Select all the components required to represent a number in IEEE floating point format.

**Solution:** Selected options in bold:

**Sign**

**Exponent**

Overflow or underflow

**Fraction**

**Q 5** Arrange the steps involved in floating point multiplication. The first step must be at the top and the last step must be at the bottom. You can assume that the result is still normalized after rounding.?

**Solution:** The steps are in the image below. We do the check for overflow/underflow after normalizing to because normalization may resolve an apparent overflow/underflow after exponent calculation.

Start

1	⋮ Add biased exponents of the two numbers
2	⋮ Subtract bias from the sum
3	⋮ Multiply the significands
4	⋮ Normalize the product, if needed
5	⋮ Check for overflow or underflow. If true, flag an exception, otherwise continue.
6	⋮ Round the significand
7	⋮ Set the sign of the result

Done