Computer Architecture

Assignment 3

174170P

Thambavita H.R.J.B

X	Y	Overflow?	Type of overflow	Justification
110000	011111	No		When the X, Y have opposite signs, their sum will never overflow.
111111	111111	No		Both X, Y have negative signs and their sum is also negative. 111111+111111=111110 Therefore their sum will never overflow.
000111<<2	000011	No		X=000111<<2= 011100 Both X, Y have positive signs and their sum is also positive. 011100+000011=011111
000111	111000>>3 (arithmetic)	No		111000>>3=111111 When X, Y have opposite signs, their sum will never overflow.

- **Fractional binary numbers**

  - **What is 1011.101_2?**

    1011.101 = 8 + 2 + 1 + ½ + 1/8
    = 11 + 5/8
    = 93/8
    = 11.625_{10}

- **<u>Examples</u>**

Fractional value	**Binary representation**	**Decimal representation**
1/8	0.001	0.125
3/4	0.11	0.75
25/16	1.001	1.5625
43/16	10.1011	2.6875
9/8	1.001	1.125
47/8	101.111	5.875
51/16	11.0011	3.1875

- **<u>IEEE floating point representation</u>**

  - **Encode the following numbers in both single precision and double precision IEEE Format**
    - 15214.0

      $15214.0 = 00111011\ 01101110_2$

      $= 1.1101101101110 * 2^{13}$

      Sign = 0

      - **<u>Single precision</u>**

        Biased exponent = 127+13 = 140

        $140 = 10001100_2$

        Normalized mantissa = 110 1101 1011 1000 0000 0000

31	30 23	22 0
0	**10001100**	**1101101 10111000 00000000**

      - **<u>Double precision</u>**

        Biased exponent = 1023 + 13 = 1036

        $1036 = 10000001100_2$

        Normalized mantissa = 1101 10110111 00000000 00000000 00000000 00000000 00000000

63	62 52	51 32
0	**10000001100**	**1101 10110111 00000000**

31			0
00000000	**00000000**	**00000000**	**00000000**

- **527.456**

  $527 = 10\ 00001111_2$
  $0.456 = 0.0111\ 0100\ 1011\ 1100\ 0110\ 1010_2$
  $527.456 = 10\ 0000\ 1111.0111\ 0100\ 1011\ 1100\ 0110\ 1010_2$
  $= 1.0000\ 0111\ 1011\ 1010\ 0101\ 1110\ 0011\ 0101\ 0 * 2^9$
  Sign = 0

  - **<u>Single precision</u>**

    Biased exponent = 127+ 9= 136
    $136 = 1000\ 1000_2$
    Normalized mantissa = 000 0011 1101 1101 0010 1111

31	30 23	22		0
0	**10001000**	**00000111**	**10111010**	**0101111**

  - **<u>Double precision</u>**

    Biased exponent = 1023 + 9 = 1032
    $1032 = 100\ 0000\ 1000_2$
    Normalized mantissa = 0000 01111011 10100101 11100011 01010011 11110111 11001110

63	62 52	51		32
0	**10000001000**	**0000**	**01111011**	**10100101**

31			0
11100011	**01010011**	**11110111**	**11001110**

- **<u>Tiny Floating Point Example</u>**

    - Assume 6-bit IEEE floating point format when k = 3 exponent bits and n=2 fraction bits
        - Calculate the Bias

          Bias= $2^{k-1} - 1 = 2^{3-1} - 1 = 3$

        - Convert the following from floating point into decimal values

0	010	01
S	k	f

3+Exp = 2
Exp= -1
$1.01 * 2^{-1} = 0.101$

$0.101 = 0.625_{10}$

0	001	00
S	k	f

3+Exp = 1
Exp = -2

$1.00 * 2^{-2} = 0.01$

$0.01 = 0.25_{10}$