ASSIGNMENT – 11

In this assignment the goal was to implement the adder for two floating point numbers. The input is taken from an input file in terms of binary representation of both the operands and the sum is written to an output file.

The whole process can be divided into steps as follows -

1. The size of the exponents of both the operands is compared and the one with the smaller operand is shifted by an appropriate value in order to match both the exponent values.

2. Both the significands are added.

3. The sum is normalized by either shifting right and incrementing the exponent or shifting left and decrementing the exponent.

4. The sum is then checked for overflow or underflow. If yes then an exception is returned else it proceeds to the next step.

5. The sum is then estimated to accomodate with in the bit size architecture and then checked once again for the normalization and overflow/underflow. If normalized then the process exits and the sum is returned.

EXCEPTION HANDLING -

1. Overflow – if exponent of sum >254 it leads to overflow exception.

2. Underflow – if exponent of sum < 1 it leads to underflow exception.

Infinity and Nan exceptions -

Inf + Inf = Inf
Inf + NaN = NaN
NaN+ other = NaN
Inf + other = Inf
Nan + NaN = NaN
Inf – Inf = NaN