

## Computer Architecture

### Lab Assignment

Total Marks = 14 (including demonstration and viva)

Due Date: 21 Dec 2021 till 23:59 Hrs. [No extension of due date is possible]

The text file containing the program needs to be submitted by the due date in LMS in "Lab Assignment". **There is no marks for only submission of the assignment.** You need to demonstrate your code during the lab evaluation class on 22 Dec 2021 or 23 Dec 2021 as appropriate, and also if require I may ask you to modify / add / replace some code as part of evaluation & viva.

Let us define our customized floating point number system (called as LPFP => Low Precision Floating Point number) in 32 bits as follows:

**Sign bit:** most significant bit (0 => the number is positive, 1=> the number is negative)

**Biased exponent:** next 15 bits [Note that Bias value is **16383**]

**Mantissa:** 16bits

All these floating point numbers are in normalized format.

Write Assembly Language program to **Add** and **Multiply** two LPFP numbers. Also write additional code / data to test these functions.

Note:

- Implementation must be modular.
- You need to write **lpfpAdd** and **lpfpMultiply** as two functions.
- Data must be taken from memory. And After computation the result has to be put into memory.
- Each function assumes that address is stored in register [r1] from where the two 32-bit lpfp numbers must be taken.
- And the result needs to be put into location pointed by register [r1] just after the input data.
- All registers (except [r1]) used inside these functions must be restored to its original value after the end of function call.