Questions on IEEE754

1)Precision

IEEE754 defines a technical standard for floating point computation. Any number has three parts in its representation

- Sign
- Exponent
- Mantissa

Sign	Exponent	Mantissa

Representation of a floating point number in IEEE754

Precision is a measure of how accurately a number can be represented. Range is a quantifier of how wide a bunch of numbers can be represented.

Precision is directly dependent on the number of bits the mantissa gets. If there are more bits designated to the mantissa, the representation of the number is very accurate/precise. If there are fewer bits, then two numbers which are close to each other in value might end up with the same representation.

IEEE 754 defines two kinds of precision: single and double. Single has 23 bits dedicated to the mantissa while double has 52.

Let's consider an example to further illustrate this:

The value of pi upto 10 decimal places is **3.14159265**

Representation in single precision

0	10000000	10010010000111111011010

Representation in double precision

0	10000000000	1001001000011111101101010011110010001101010011110001

On converting these back to decimal values,

Single precision=3.1415925

Double precision=3.14159265

It can be seen that there is a loss of accuracy while storing pi in single precision format whereas double precision retains the entire number. This is due to the number of bits allocated to the mantissa.

2)Normal and subnormal values

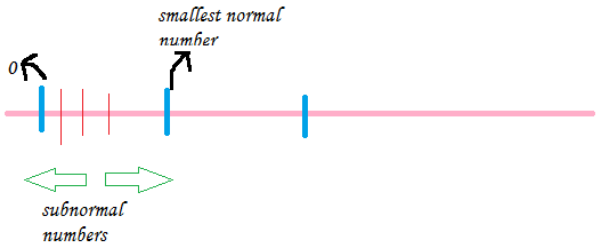
According to IEEE754, there is a range of numbers that can be represented. By definition, normal values are those which fall into the prescribed range of a floating point format. In contrast, subnormal values are those whose exponent is zero. Subnormal values are those numbers whose value is lesser than the smallest normal number.

Purpose of subnormal values

According to IEEE754, the magnitude of the smallest possible number is b^{emin} where emin is the minimum value the exponent can take.

Suppose this value is 4.This means that the smallest number that can be represented is -2^4 which is 0.0625.Now, if we had to subtract 0.03125 from 0.0625, the result would be zero if we had only normal numbers. This is because both these numbers would be represented by the same value and therefore, the result of subtraction would be zero. Such conditions are termed underflows. This is undesirable. To avoid this, the concept of subnormal numbers was introduced. These numbers are permitted to have a leading zero in their mantissa. By doing this, the gap between zero and the smallest number that can be represented is brought down.

An illustration depicting normal and subnormal numbers



3)Rounding methods

IEEE754 defines 5 rounding rules. The first two round to the nearest value whereas the others are called direct roundings.

- Round to nearest, ties to even

  This method rounds to the nearest value. If the number falls exactly midway, then the value with LSB 0(even) is chosen.

  Ex:13.5—>14 12.5→12

- Round to nearest, ties away from zero

  This method also rounds to the nearest value; but in case of a tie, positive numbers are rounded off to the closest number above whereas negative numbers are rounded to the closest number below.

  Ex:12.5—>12 -11.5→-12

- Round up, or round toward plus infinity

This is also called ceiling, where the rounding is always to a larger number.

Ex:11.5→12 -11.5->-11

- Round down, or round toward minus infinity

This is also called floor, where the numbers are always rounded to a smaller value.

Ex:11.5→11 -13.5→-14

- Round toward zero, or chop, or truncate

This is called truncation where the output chosen is always closer to 0.

Ex.11.5→11 -10.5→-10

References

1) https://en.wikipedia.org/wiki/IEEE_754
2) https://en.wikipedia.org/wiki/Denormal_number
3) https://en.wikipedia.org/wiki/Normal_number_(computing)
4) https://en.wikipedia.org/wiki/IEEE_754#Rounding_rules
5) http://www.keil.com/support/man/docs/armlib/armlib_chr1358938950865.htm