Date

*) Introduction to Computer Science Sheet #05

*) Problem 5.1:

a) The range of integer numbers that can be represented by b3n6 is given by b^n i.e

$3^6 = 729$.

Hence, to find the smallest and largest numbers that can be represented we would divide this value by 2.

So, $\frac{729}{2} = 364.5$

Therefore, we can say that the largest possible number represented is +364 and smallest possible being -364. This is because 0 takes the remaining number space in the range of b3n6.

b) To represent -1:

+1 in b3n6 would be:

3^5	3^4	3^3	3^2	3^1	3^0
243	81	27	9	3	1
0	0	0	0	0	1

Now, using $a_i' = (b-1) - a_i$,

$000001 \rightarrow 222221$

Now, adding 1 to it =

$$\begin{array}{r} 222221 \\ +\ 000001 \\ \hline 222222_3 \end{array}$$

$\Rightarrow$ this represents -1.

Date

To represent -99:

+99 in b_3n6 would be:

3^5	3^4	3^3	3^2	3^1	3^0
243	81	27	9	3	1
0	1	0	2	0	0

Now using $a_i' = (b-1) - a_i$

$010200 \rightarrow$ ~~212002~~
212022.

Now adding 1 to it:

$$\begin{array}{r} 212022 \\ 000001 \\ \hline 212100 \end{array}$$

→ This represents -99.

c) Adding the values of -1 and -99:

$$\begin{array}{r} 222222 \\ +\ 212100 \\ \hline 1212022 \end{array}$$

1212022_3

As it is overflow, the output in b3n6 would be 212022_3 only.

Date

To convert the result back to the decimal number system:

$$\begin{array}{r} 212022_3 \\ -\ 000001 \\ \hline 212021 \end{array} \longrightarrow 010201_3$$

The decimal representation then is 100_{10}.

Date

Problem 5.2:

a) To represent 321.123 in a single precision floating point number, we first have to compute the bit for the sign. As the number is positive, the sign bit will be 0 in this case.

Secondly, we'll compute the conversion of the part before the decimal point i.e 321.
To do so, we would mod 2 the value till it reaches 0.
This is as follows:

321 mod 2	= 1	1
160 mod 2	= 0	01
80 mod 2	= 0	001
40 mod 2	= 0	0001
20 mod 2	= 0	00001
10 mod 2	= 0	000001
5 mod 2	= 1	1000001
2 mod 2	= 0	010000001
1 mod 2	= 1	101000001

Hence, 321 is represented as
101000001

Date

Now to represent the part after the decimal point we would multiply it with 2 and yield the integer part until it reaches 0.0 or a pattern repeats itself. This is as follows:

0.123	× 2 = 0.246	0
0.246	× 2 = 0.492	0
0.492	× 2 = 0.984	0
0.984	× 2 = 1.968	1
0.968	× 2 = 1.936	1
~~0.872~~	~~× 2 = 1.744~~	~~1~~
0.936	× 2 = 1.872	1
0.872	× 2 = 1.744	1
0.744	× 2 = 1.488	1
0.488	× 2 = 0.976	0
0.976	× 2 = 1.952	1
0.952	× 2 = 1.904	1
0.904	× 2 = 1.808	1
0.808	× 2 = 1.616	1
0.616	× 2 = 1.232	1
0.232	× 2 = 0.464	0
0.464	× 2 = 0.928	0
0.928	× 2 = 1.856	1
0.856	× 2 = 1.696	1
0.696	× 2 = 1.392	1
⋮		

As the number is constantly growing and exceeds the limit of the bits of the mantissa we would stop here.

Date

Finally, to represent both ~~312.1~~ 321.123 together it would be

101000001.00011111011110

We can now remove the most significant bit '1' as when normalized we move up the decimal point to right after the '1' in every case. This is the IEEE 754 convention, and we assume that the first digit is always 1.

Thus, to normalize it we would multiply by 2^8 as follows for the exponent =

mantissa = 1.0100000100011111011110

exponent = for bias shift =

$8 + 127$

$= 135$

this would be =

135 mod 2	=	1
67 mod 2	=	11
33 mod 2	=	111
16 mod 2	=	0111
8 mod 2	=	00111
4 mod 2	=	000111
2 mod 2	=	0000111
1 mod 2	=	10000111

Therefore, our final representation would be =

S	exponent	mantissa (23 bits)
0	10000111	0100000100011111011110

//

Date

b) The decimal fraction stored in the single precision floating point number is as follows:

$$00011111\,0111110$$

This is the binary representation of

$$0.1229858398$$

The absolute error then is =

$$0.123 - 0.1229858398$$
$$= -0.0000141602$$

$$\approx 1.41602 \times 10^{-5}$$