

Homework #1 Solution

(Paper Based Part)

September 26, 2004

In all of the following, be sure to show all your work (including the tables for the multiplication and division methods when performing radix conversions)

Problem 1. (10 points each)Convert each of the following numbers from decimal into 32-bit IEEE floating point format. Give your final answer as an eight digit hexadecimal number.

- 13.3125

a) 13 => 1101

Old	Old/2	Old%2
13	6	1
6	3	0
3	1	1
1	0	1

b) .3125 => .0101

Old	Bit	New
.3125	0	.625
.625	1	.25
.25	0	.5
.5	1	0

c) 13.3125 => 1101.0101 => 1.1010101 * 2³

d) Sign = 0
Exp = 3 + 127 = 130 = 10000010
Mantissa = 1010101

Result:
0 10000010 101010100000000000000000
0100 0001 0101 0101 0000 0000 0000 0000
4 1 5 5 0 0 0 0 (Hex)

- 0

Result: (a special case)
0000 0000 0000 0000 0000 0000 0000 0000
0 0 0 0 0 0 0 0 (Hex)

- -4.5

```

a) -4 => 100
b) .5 => .1
c) -4.5 => 100.1 => 1.01 * 2^2
d) Sign    = 1
    Exp    = 2 + 127 = 129 = 10000001
    Mantissa = 01

Result:
1 10000001 010000000000000000000000
1100 0000 1010 0000 0000 0000 0000 0000
C    0    9    0    0    0    0    0    0    (Hex)

```

- 0.7

```

a) 0 => 0
b) .7 => .101100110011001100110011
    Old Bit New
    .7 1 .4
    .4 0 .8
    .8 1 .6
    .6 1 .2
    .2 0 .4
    .4 0 .8
    .8 1 .6
    .6 1 .2
    .2 0 .4
    .4 0 .8
    .8 1 .6
    .6 1 .2
    .2 0 .4
    .4 0 .8
    .8 1 .6
    .6 1 .2
    .....
c) 0.7 => .101100110011001100110011 => 1.0110011001100110011 * 2^-1
d) Sign    = 0
    Exp    = -1 + 127 = 126 = 01111110
    Mantissa = 01100110011001100110011

Result:
0 01111110 01100110011001100110011
0011 1111 0011 0011 0011 0011 0011 0011
3    f    3    3    3    3    3    3    (Hex)

```

Problem 2. (10 points each) Convert each of the following numbers from 32 bit floating point format into ordinary human-readable decimal.

- 3f800000

```
0011 1111 1000 0000 0000 0000 0000 0000
0 01111111 000000000000000000000000
```

```
Sign   = 0
Exp    = 01111111 => 0
Mantissa = 0
```

$1.0 * 2^0 = 1.0$

- 42148000

```
0100 0010 0001 0100 1000 0000 0000 0000
0 10000100 001010010000000000000000
```

```
Sign   = 0
Exp    = 10000100 => 5
Mantissa = 00101001
```

$1.00101001 * 2^5 = 100101.001$

$37 + 1/8 = 37.125$

- c22d4000

```
1100 0010 0010 1101 0100 0000 0000 0000
1 10000100 010110101000000000000000
```

```
Sign   = 1
Exp    = 10000100 => 5
Mantissa = 010110101
```

$1.010110101 * 2^5 = 101011.0101$

$-45 + 5/16 = -45.3125$

- bd800000

```
1011 1101 1000 0000 0000 0000 0000 0000
1 01111011 000000000000000000000000
```

```
Sign   = 1
Exp    = 01111011 => -4
Mantissa = 0
```

$-1.0 * 2^{-4} = -.0001$

$-1/16 = -.0625$

Problem 3. (20 points each) Perform the following floating point operations using the algorithms we described in class.

- $3f800000 + 3f800000$

a) expand

```
3f800000 => 0011 1111 1000 0000 0000 0000 0000 0000
           => 0 01111111 000000000000000000000000
Sign      = 0
Exp       = 01111111 = 0
Mantissa  = 0
3f800000 => 1.0 * 2^0
```

b) reduce to the same exponent (already are)

c) add the mantissa

```
  1.0
+1.0
-----
 10.0
```

d) renormalize

```
10.0 * 2^0 = 1.0 * 2^1
Sign       = 0
Exp        = 1 + 01111111 = 10000000
Mantissa   = 0
```

e) reconstruct the answer in IEEE floating point format

```
0 10000000 000000000000000000000000
0100 0000 0000 0000 0000 0000 0000 0000
4   0   0   0   0   0   0   0   0   (Hex)
```

- $41420000 + c15a0000$

a) expand

```
41420000 => 0100 0001 0100 0010 0000 0000 0000 0000
           => 0 10000010 100001000000000000000000
c15a0000 => 1100 0001 0101 1010 0000 0000 0000 0000
           => 1 10000010 101101000000000000000000
```

b) reduce to the same exponent (already are)

c) add the matissa

```
  1.101101
-1.100001
-----
 0.001100  => 1.1 * 2^(-3)  (negative result)
```

d) remormalize

```
Exp = 10000010
     -00000011 (this is -3)
     -----
           01111111
```

e) reconstruct the answer in IEEE floating point format

```

Sign = 1
Matissa = 1.1
Exp     = 01111111

1 01111111 100000000000000000000000
1011 1111 1100 0000 0000 0000 0000 0000
b   f   c   0   0   0   0   0   0   (Hex)

```

• 40000000 * 40000000

a) expand

```

40000000 => 0100 0000 0000 0000 0000 0000 0000 0000
=> 0 10000000 000000000000000000000000

```

b) multiply the matissa

```

  1.0
* 1.0
----
  1.0

```

c) add the exponents and fit into IEEE exponent format

```

10000000
+ 10000000
-----
100000000
- 01111111
-----
100000001

```

d) renormalize (already normalized)

e) reconstruct the answer in IEEE floating point format

```

Sign = 0
Exp   = 10000001
Matissa = 1.0

0 10000001 000000000000000000000000
0100 0000 1000 0000 0000 0000 0000 0000
4   0   8   0   0   0   0   0   0   (Hex)

```

• 422e0000 * be000000

a) expand

```

422e0000 => 0100 0010 0010 1110 0000 0000 0000 0000
=> 0 10000100 010111000000000000000000
be000000 => 1011 1110 0000 0000 0000 0000 0000 0000
=> 1 01111100 000000000000000000000000 (Negative number !)

```

b) multiply the matissa

```

  1.010111
* 1.0
-----
  1.010111

```

(Result is negative)

c) add the exponents and fit into IEEE exponent format

```
10000100
+01111100
-----
100000000
- 01111111
-----
100000001
```

d) renormalize (already are)

e) reconstruct the answer in IEEE floating point format

```
Sign = 1
Exp  = 10000001
Matissa = 1.0100111

1 10000001 010011100000000000000000
1100 0000 1010 0111 0000 0000 0000 0000
c   0   a   e   0   0   0   0   0   (Hex)
```

• 42c80000 / 40a00000

a) expand

```
42c80000 => 0100 0010 1100 1000 0000 0000 0000 0000
=> 0 10000101 100100000000000000000000
40a00000 => 0100 0000 1010 0000 0000 0000 0000 0000
=> 0 10000001 010000000000000000000000
```

b) divide the matissa

```
      1.01
-----
1.01 ) 1.1001
      1.01
      ---
        1
        --
         10
         ---
          101
          ---
           101
           ---
            0
```

c) subtract the exponents and fit into IEEE exponent format

```
10000101
-10000001
-----
00000100
+ 01111111
-----
10000011
```

d) renormalize (already are)

e) reconstruct the answer in IEEE floating point format

```

Sign = 0
Exp  = 10000011
Matissa = 1.01

0 10000011 010000000000000000000000
0100 0001 1010 0000 0000 0000 0000 0000
4   1   a   0   0   0   0   0   0   (Hex)

```

• 41700000 / c1100000

a) expand

```

41700000 => 0100 0001 0111 0000 0000 0000 0000 0000
           => 0 10000010 111000000000000000000000
c1100000 => 1100 0001 0001 0000 0000 0000 0000 0000
           => 1 10000010 001000000000000000000000 (Negative number!)

```

b) divide the matissa

```

          1.10101010101.....
-----
1001)1111
   1001
   ----
     1100
     1001
     ----
      110
      ----
       1100
       1001
       ----
        110
        .....

```

c) subtract the exponents and fit into IEEE exponent format

```

    10000010
  - 10000010
  -----
    00000000
  + 01111111
  -----
    01111111

```

d) renormalize (no need in here)

e) reconstruct the answer in IEEE floating point format

```

Sign = 1
Exp  = 01111111
Matissa = 1.10101010101.....

1 01111111 1010101010101010101010101
1011 1111 1101 0101 0101 0101 0101 0101
b   f   d   5   5   5   5   5   5   (Hex)

```