Q: What connects Seattle and Bellevue?
A: The Evergreen Floating Point Bridge.

Floating Point Arithmetic

Most scientific and engineering computations require "decimal" arithmetic, i.e. numbers containing a decimal point. Floating point is the computer implementation of "real" arithmetic with limited precision. Until recently, only the largest computers had floating point hardware as standard equipment.

Terminology

Scientific Notation: 3.1557×10^{9}, $3.14.16 \times 10^{0}$, 3.1557×10^{-9}

Normalized Number: 31.557×10^{8} ---> 0.31557×10^{10}

Decimal Fractions	
0.1_{10}	$1*10^{-1}$
0.44_{10}	$44*10^{-2}$
0.612_{10}	$612*10^{-3}$

Binary Fractions		
0.1_2	$1*2^{-1}$	1/2
0.01_2	$1*2^{-2}$	1/4
0.101_2	$5*2^{-3}$	5/8

- General Form of Floating Point: $1.\text{fffffffff} \times 2^{eee}$
- Constituents are: sign, significand or mantissa and exponent

s	exponent	mantissa
1 bit	8 bits	23 bits

Further Floating Facts

Floating point is well understood: IEEE 754 FP Standard
Single precision -- one word representation of fp
Double precision -- two word representation of fp
Range --
  Single: 2.0×10^{-38} through 2.0×10^{38}
  Double: 2.0×10^{-308} through 2.0×10^{308}
MSB of normalized mantissa not represented: 24, 53 bits
Zero is represented as 000...0, i.e. it has no implied MSB
FP has the property that when a<b as signed magnitude numbers then a<b as floating point numbers

Biased Representation

`0 01111111 00000000000000000000000`   **1.0×2^{-1}**
`0 00000001 00000000000000000000000`   **$1.0 \times 2^{+1}$**

- Signed exponents would complicate comparisons
- In biased notation the most negative number is $000...0_2$ and the most positive is $111...1_2$
- Since the single precision exponent field is 8 bits, allowing 256 different configurations, the bias for sp fp is 127
  - +2 is presented as 2+127 = 129 = 1000 0001
  - -2 is represented as -2+127 = 125 = 0111 1101
- The bias for double precision is 1023
- The formula: $(-1)^{sign}*(1+\text{mantissa})*2^{(\text{exponent-bias})}$

Example Representations

- Find floating point for 5.125
  - $5.125 = 5 + 0.125 = 5 + 1/8 = 5 + 1*2^{-3}$
  - $= 101_2 + .001_2 = 101.001$
  - Normalize: 101.001_2*2^{0} ---> $1.01001*2^{2}$
  - Thus $5.125 = (-1)^{0}*1+.0100\ 1000\ 0... * 2^{129}$

`0 10000001 01001000000000000000000`

- In reverse, what floating point number is
  - $(-1)^{1}*(1.0111\ 000\ 0...)*2^{130}$
  - In birary scientific notation it is $-1.0111*2^{3}$
  - Reducing the exponent to 0 yields -1011.1_2
  - $= -(11_{10} + 2^{-1)}) = -(11_{10} + 1/2) = -11.5_{10}$

Multiplying Floating Point Numbers

- Recall that fp is scientific notation, so arithmetic is logarithmic
  - Add the exponents (reduce by the bias), multiply the mantissas and renormalize if needed

```
0.75 x 24 = 0.11 x 11000
1.1x2^-1 times 1.1x2^4
Add exponents: -1 + 4 = 3
Multiply fractions:
        1.1
        1.1
        1 1
      1 1
    1 0.0 1
Result: 10.01x2^3 = 10010
   or  1.001x2^4 normalized
```

```
0.75 x 24 = 0.11 x 11000
1.1x2^126 times 1.1x2^131
Add exponents:
    126+131 = 257-127 = 130
Multiply fractions:
        1.1
        1.1
        1 1
      1 1
    1 0.0 1
Result: 10.01x2^130 = 1.001x2^131
```

Adding Floating Point Numbers

- Requires that the binary points be aligned
- Equivalent to having the same exponent ...

  shift the mantissa of the smaller right, raising its exponent

$1.000*2^{-1} + 1.011*2^{2}$ (0.5 + 5.5)
Shift smaller right: $1.000x2^{-1}=0.100x2^{0}=0.0100x2^{1}=0.001x2^{2}$
Add: $1.011*2^{2} + 0.001*2^{2} = 1.100*2^{2}$
Renormalize: $1.100*2^{2}$ ... it's OK
Result: $1.100*2^{2}$

Floating Point Instructions

- There are 32 fp registers: $f0, $f1, ... $f31
  - The even numbered registers are used for sp
  - An even/odd pair is used for dp, with the odd numbered register holding the lsb mantissa bits
- Special load/store instructions move fp data to/fro mem
  - `l.s, s.s, l.d, s.d`
- Arithmetic operations (R-type) come in sp/dp forms
  - `add.s, add.d, sub.s, sub.d, mul.s, mul.d`
- Comparisons make direct tests and set a condition bit
  - `c.le.s, c.lt.s, c.eq.s, c.ne.s, c.gt.s, c.ge.s`
  - `c.le.d, c.lt.d, c.eq.d, c.ne.d, c.gt.d, c.ge.d`
- Branch if true, `bc1t`, and branch if false, `bc1f`

To Infinity and Beyond

- IEEE 754 reserves certain representations for extreme conditions:

Single		Double		
Exp	Signif	Exp	Signif	Meaning
0	0	0	0	Zero
0	nonzero	0	nonzero	+/- unnormal
1-254	anything	1-2046	anything	+/- floating p
255	0	2047	0	+/- infinity
255	nonzero	2047	nonzero	NaN