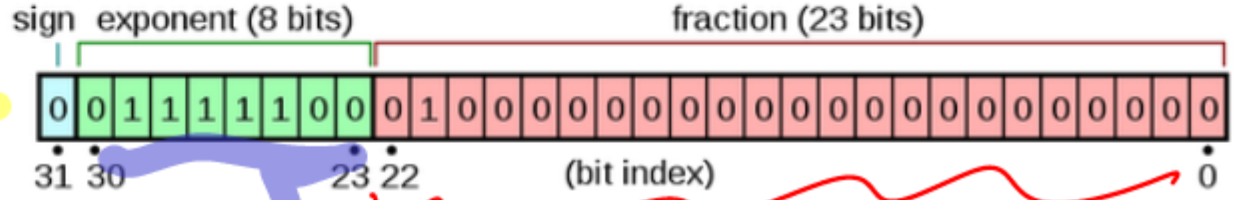
An example of a layout for 32-bit floating point is



and the 64 bit layout is similar.

the first digit is not stored

Question: How to represent 0?

① Want all bits equal zero to mean zero

$e = 0 \qquad m = 0 \qquad s = 1$

recall the exponent is actually $e-127$ so one can store negative exponents.

$$1.01 \times 2^{e-127}$$

Store integers on a computer...

$$b_{17} \cdots b_2 b_1 b_0 = \sum_{k=0}^{17} b_k 2^k \qquad b_i \in \{0,1\}$$

only positive integers...

How to store negative integers?

① Sign bit. (1's complement)

② 2's complement

③ bias or offset

Since $e=0$ corresponds to 2^{-127} exponent maybe we don't need that exponent at all and thus 2^{-126} is the smallest exponent that actually exists...

When $e=0$ this is then a special case to store

0, not-a-number, ∞, $-\infty$, etc.

Sorry, at this point I forgot to save the lecture notes and they where lost when the computer powered off.

I'll try to recreate what I remember discussing:

STEP 3

First I talked about different types of error:

1. Initial Error – this is the error in the inputs to a calculation
2. Propagated Error – this is how the initial error affects the answer.
3. Generated Error – this is error created during the calculation by rounding
4. Accumulated Error

   = Propagated + generated error.

There are two different ways to measure error:

absolute and relative.

Let $x \in \mathbb{R}$ and x^* be an approximation of x.

absolute error $\quad e_{abs} = |x - x^*|$

A number correct to n decimal places has

$$e_{abs} \leq 0.5 \times 10^{-n}$$

Example: Let $x \in \mathbb{R}$ and let $x^* = 4.126$ be an approximation of x correctly rounded to the digits shown.

The largest x that rounds to x^* is 4.1265

The smallest x that rounds to x^* is 4.1255

Thus, we know $x \in [4.1255, 4.1265]$. In other words

$$x = x^* \pm 0.0005 = 4.126 \pm 0.0005.$$

A correctly rounded number is one which is closest to the number being rounded.

When rounding it can happen that there are two numbers which are equally close to the number being rounded.

For example:

3.125 and 3.126 are equally close to 3.1255

In the case of a tie, chose the approximation whose final digit is even. In this case, we would round

$$(3.1255)^* = 3.126$$

since 6 is even.

Example: Let $x \in \mathbb{R}$ and let $x^* = 8.13$ be an approximation of x correctly rounded to the digits shown.

Thus $x = 8.13 \pm 0.005$ and in particular we know

$$x \in (8.125, 8.135)$$

note the endpoints aren't included because of the round to even on a tie rule.

Note also the notation 8.13 ± 0.005 doesn't explicitly indicate whether the endpoints are included or not.

relative error

$$e_{rel} = \frac{e_{abs}}{|x|} \leq \frac{e_{abs}}{|x^*| - e_{abs}} \approx \frac{e_{abs}}{|x^*|}$$

The above assumes the absolute error is small compared to the size of x and its approximation.

unless we a being very careful we will interchangeably use

$$\frac{e_{abs}}{|x|} \qquad \frac{e_{abs}}{|x^*|}$$

based on which is more convenient.

A decimal number correct to n significant digits has

$$e_{rel} \leq 5 \times 10^{-n}$$

Note p implies q is different than q implies p

In particular, if $e_{rel} \leq 5 \times 10^{-n}$ then one can not immediately conclude that the number is correct to n significant digits.

Suppose $x \in \mathbb{R}$ and $x^* \approx 2.31$ be an approximation for x correctly rounded to the digits shown.

Find a bound on the relative error...

Since $x = x^* \pm 0.005$ then $x \in (2.305, 2.315)$ and at most

$$e_{abs} = |x - x^*| \lesssim 0.005.$$

Now

$$e_{rel} = \frac{e_{abs}}{|x|} \approx \frac{e_{abs}}{|x^*|} = \frac{0.005}{2.31} \approx 0.0021645\ldots$$

```
julia> 0.005/2.31
0.0021645021645021645
```

Note that since 2.31 is good to $n = 3$ significant digits, then we know

$$e_{rel} \lesssim 5 \times 10^{-3} = 0.005$$

A decimal number correct to n significant digits has

$$e_{rel} \leq 5 \times 10^{-n}$$

The fact that our more precise estimate

$$0.0021645\ldots \lesssim 0.005$$

shows consistency in these bounds..

Now, show that q does not imply p, that is, $e_{rel} \leq 5\times10^{-n}$ does not necessarily imply x^* is good to n significant digits.

Suppose $e_{rel} \leq 5\times10^{-3}$ and $x^* = 9.38$ then

$$\frac{|x - 9.38|}{|x|} \leq 5\times10^{-3} \quad \text{implies}$$

assuming (as is reasonable) that $x > 0$ that

$$|x - 9.38| \leq (5\times10^{-3})\, x$$

or

$$-(5\times10^{-3})\, x \leq x - 9.38 \leq (5\times10^{-3})\, x$$

Thus

$$9.38 \leq (1 + 5\times10^{-3})\, x \quad \text{and} \quad (1 - 5\times10^{-3})\, x \leq 9.38$$

equivalently

$$\frac{9.38}{1+5\times10^{-3}} \leq x \quad \text{and} \quad x \leq \frac{9.38}{1-5\times10^{-3}}$$

```
julia> 9.38/(1+5e-3)
9.333333333333336
```

```
julia> 9.38/(1-5e-3)
9.427135678391961
```

Consequently

$$x \in [9.3333\ldots, 9.42713\ldots]$$

It follows x is only known to 1 significant digit.