September 2, 2016

HW DUE
Floating point

In this class:

- *Understand the need for floating point arithmetic and some alternatives.*
- *Understand how the computer represents floating point numbers and how it manipulates them.*

Next class

Floating point mathematics
Floating Point
G&C – Chapter 5

Next next class

QUIZ and Floating point Math
G&C – Chapter 5

Matlab demo of the following slides

Floating point

1/3 = 0.3333333333333333

2/3 = 0.6666666666666666

\+ = 1.0000000000000000

3/10 = 0.2999999999999999

4/10 = 0.4000000000000002

\+ = 0.6999999999999996

Computers can’t subtract ????

```
>> x = 1e18;
>> x
x =
   1.0000e+18
>> y = 1e18;
>> y = y + 1;
>> y - x
ans =
     0
```

Shouldn’t this be equal to 1?

Computers approximate!

```
>> x = 1e18;
>> y = 1e18;
>> x == y
ans =
    1
>> y = y + 1;
>> x == y
ans =
    1
>> explain_double(x)
IEEE 754 Double precision floating point representation (64-bits)
   sign = 0 (1 bit)
   exp  = 10000111010 (11 bits)                              (52 bits)
   frac = 1011110000010110110101100111010011101100100000000000
   val  = 1.000000000000000e+18
```

Alternatives to floating point

(From Nick Trefethen) Exact (rational) arithmetic, we want the roots of

$$p(x) = x^5 - 2x^4 - 3x^3 + 3x^2 - 2x - 1\,.$$

No “analytical” formula. (Galois 1820s)

Alternatives to floating point

Using Newton's Method to solve a polynomial

$$x^{(0)} = 0\,,$$

$$x^{(1)} = -\frac{1}{2}\,,$$

$$x^{(2)} = -\frac{22}{95}\,,$$

$$x^{(3)} = -\frac{11414146527}{36151783550}\,,$$

Alternatives to floating point

Using Newton's Method to solve a polynomial

$$x^{(4)} = -\frac{437115663193076384403256764909499867587929989600855 36}{13863433279008761611840812755838900332126896609091862 5},$$

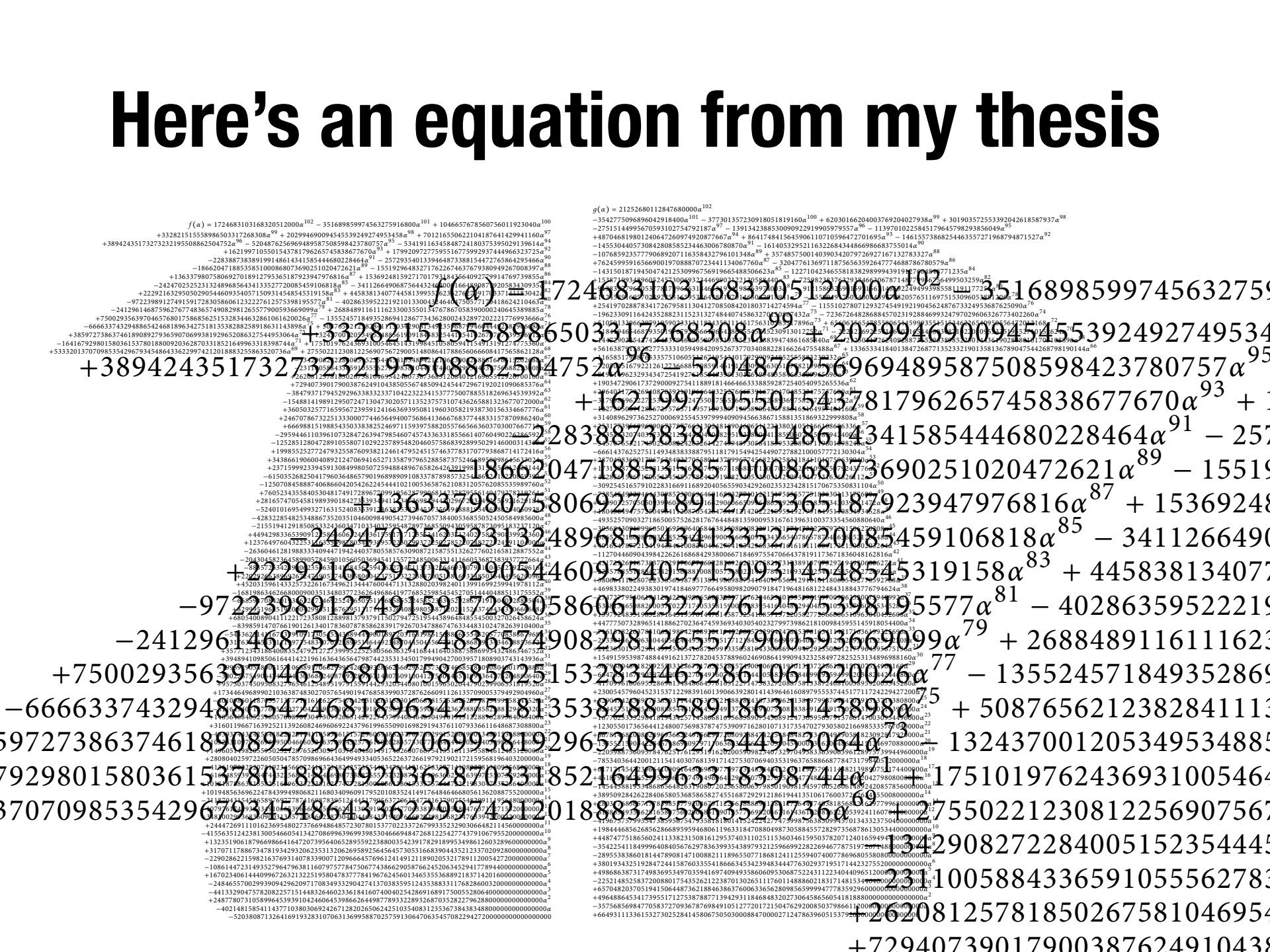
$$x^{(5)} = -\frac{724391479176820176129001381878925973035003883604754393117804119434357926010580274469629 9}{229746023731575873333990816664320035147759847208021088660066874783249488750988451982247 97}$$

$$\frac{228820641845856700177035519963166516115963436345627352999213086646631394057674120528755 38}{582289844718084679815362215689722609358654953259225717929917685478944495195182168763169 31}$$

$$\frac{201240642484300698212354536105198706894715223176068754569028985198376505504345452967792 1}{568370465908144002495419674804116675018139752278347161906687414800535564210785107754125 0}.$$

20 iterations produce a 16 terabyte file

Here's an equation from my thesis



Exact arithmetic doesn’t scale.

Fixed point

Use exact integer arithmetic. But keep a fixed decimal place.

e.g. compute with “cents” but report “dollars”

It’s hard to control scale, but for some applications, it’ll work great!

Uses of fixed point

Taxes!

Finance software (e.g. GNU Cash)

Many MP3 decoders.

Doom (the game)

What is and why floating point?

Floating point representations are like scientific notation.

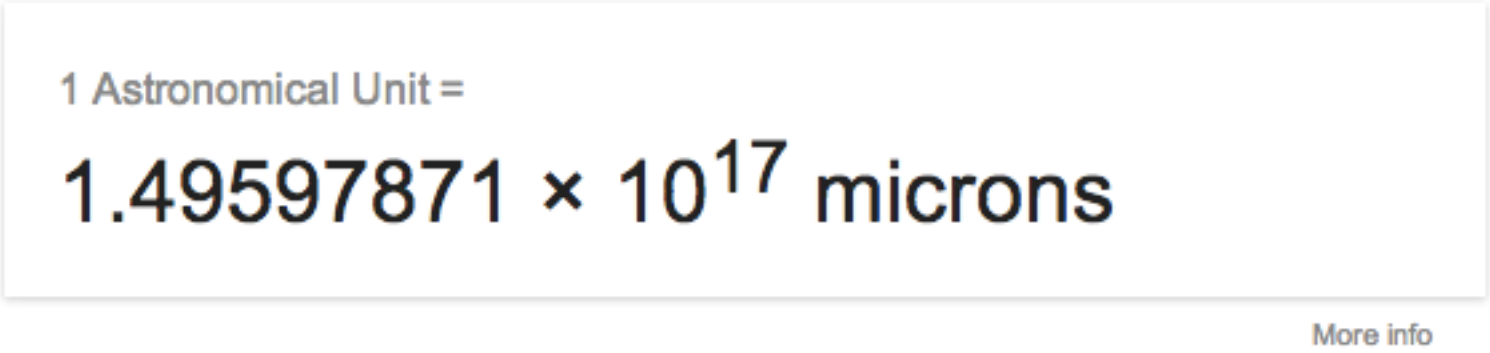
$$\text{number} = \text{significant digits} \times \text{base}^{\text{exponent}}$$

$$12,342.1 = 1.23421 \times 10^4$$

It's a compromise!

A sense of floating point scale.

The floating point numbers we'll use have around 16 significant digits.



That's enough of a dynamic range to (almost) represent a 10 micron particle between here and the sun.

Floating point arithmetic

Is $x^2 - y^2$ or $(x + y)(x - y)$ better?

How do I evaluate the roots of a quadratic?
$ax^2 + bx + c = 0$

$$f(x) = (x - t_1)(x - t_2) \cdots (x - t_k)$$
$$f'(x) = \sum_{i=1}^{n} \frac{f(x)}{x - t_i}$$

Floating point lectures

1) Mechanics of floating point (Section 5.3)

2) Mathematics of floating point (Section 5.5)

3) Study of floating point properties (Misc. ex)