Functions

Let's compare + contrast with things you learned in math class.

Math Class	C++
$f: \mathbb{R} \to \mathbb{R}$ (inputs → outputs)	`double f(double);` (outputs ← inputs)
No "side-effects" if $f(2) = 4$ now, $f(2)$ will still $= 4$ later.	Functions might have some global state.
No need to specify "how", only "what" a function does.	Always have to specify how.

```
int g = 0; // <- global var!
int f(int x)
{
  g += x;
  return g;
}
```

E.g. $f: x \mapsto x^5$

$\sin(x) =$ ratio of opposite / hypotenuse in a right triangle...

$$f(x) = \begin{cases} 1 & \text{if } x \text{ prime} \\ 0 & \text{else} \end{cases}$$

Interesting questions: for a particular function, (like factoring, for example) what are the fundamental limits to how well a program can compute the outputs? (how fast?, how much memory?)

Random question: how many functions are there from finite sets $A \longrightarrow B$?

Say $A = \{a_1, \ldots a_n\}$, $B = \{b_1, \ldots, b_m\}$.

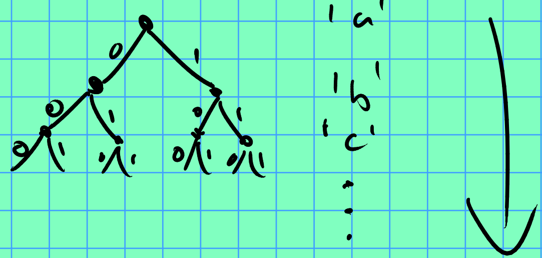
$$\left.\begin{array}{lll} \#\text{ of choices for} & f(a_1)? & m. \\ \text{" \qquad "} & f(a_2)? & m. \\ \vdots & & \vdots \\ & & m. \end{array}\right\} n \text{ times}$$

So # distinct functions from $A \to B$ is $m^n = |B|^{|A|}$.

Observation: even for relatively small A, B (e.g. $A =$ 20-character strings, $B = \{0,1\}$) the space of functions is huge.

So large, in fact, that most functions cannot have a nice description! It's interesting to find out which useful functions do.

Say we have bool f(char).



each "leaf" corresponds to a distinct function.

Random reading for those that are interested: Wiki: "The Halting Problem".