Functions

Math class:

① $f: \mathbb{R} \to \mathbb{R}$ // define domain + range / codomain

$$f(x) = x^3 - 1$$

C++

① `double f(double);` // "prototype"

②
```
double f(double x)
{
    return x*x*x - 1;
}
```

Note: in mathematics, sometimes it suffices to simply say what a function does w/o saying how:

$$p(n) = \begin{cases} 1 & \text{if } p \text{ is prime} \\ 0 & \text{else} \end{cases}$$

In programming, we must provide the "how" explicitly.

Aside: not all functions can have a concise description of "how" to compute their outputs.

Say A, B finite sets. How many functions are there from $A \to B$?

What information is necessary to define

an arbitrary function?

$A = \{a_1, a_2 \dots a_m\}$, $B = \{b_1, \dots b_n\}$.

Any function is defined by the list

$$\begin{array}{cccc} f(a_1), & f(a_2), & \dots & f(a_m) \\ \rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} & & \rotatebox{90}{$\in$} \\ B & B & & B \\ & & & \\ |B| & |B| & \dots & |B| \\ \| & \| & & \| \\ n & n & & n \end{array}$$

So, total # of distinct functions

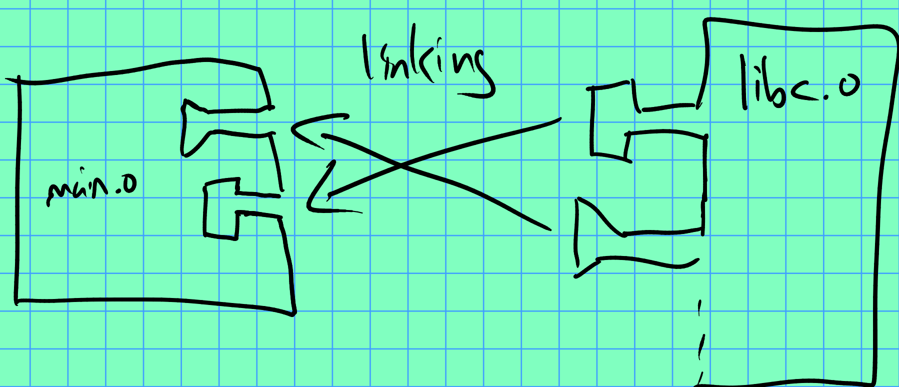
$$= n^m = |B|^{|A|}$$

So even if $\underset{64 \text{ bits}}{\text{long int}} \longrightarrow \text{bool}$

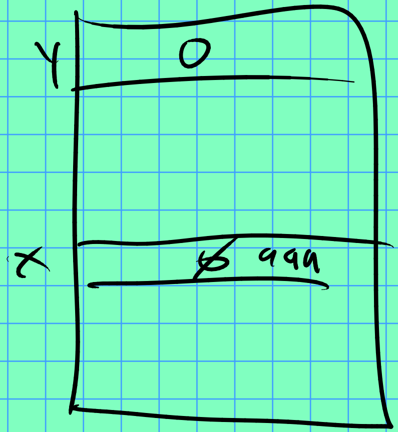
we would have $2^{2^{64}}$

Exercise: show not every function can have a concise description (smaller than $\log|B| \cdot |A|$)

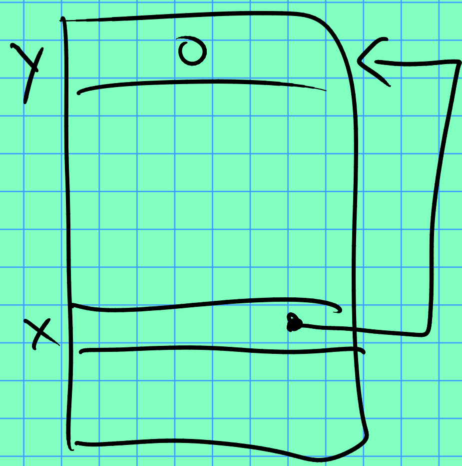
Aside: compiling vs. linking:

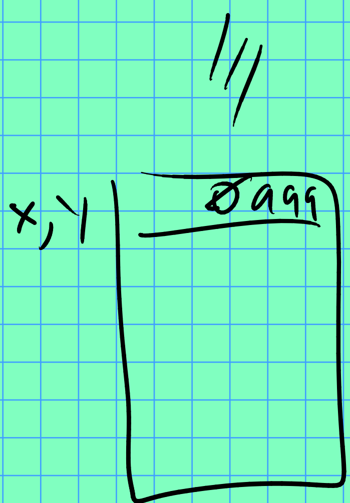


By value:



By Reference:





(Reference, alt. view)