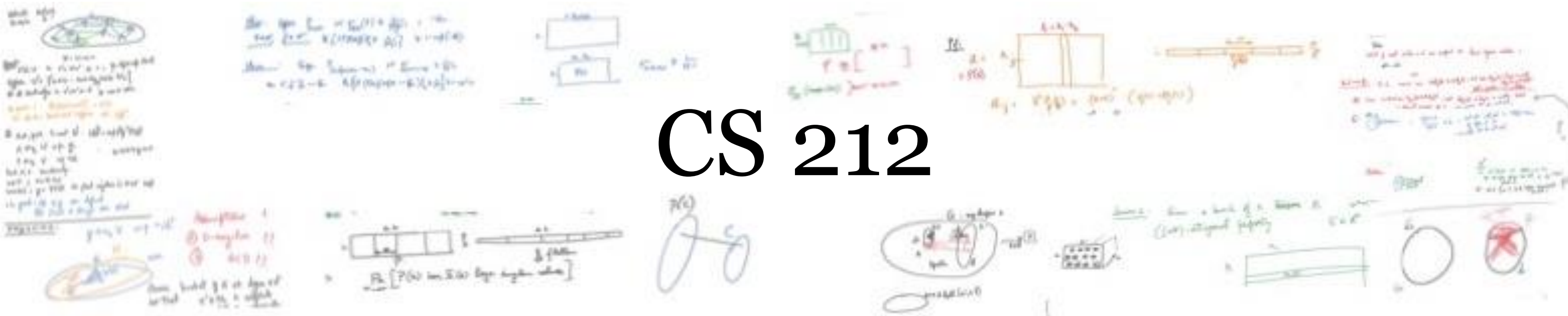


CS 212

Mathematical Foundations of Computer Science

Lecture 7: Mappings, Growth of Functions

Announcements

- Homework 2 is out. See Canvas Syllabus page for PS2.pdf
- To be submitted on Crowdmark by Oct. 11 (Tues) night (assgt will go up on Crowdmark later tonight)

- Discussion sections Wed and Thurs at 5pm.

- For any enquiries related to the course logistics (e.g., homework submissions), please send a private post on Piazza with the instructors

  (individual emails to me could get lost.)

Binary Relation

Binary relation: $a\ R\ b$ or $a \sim_R b$ means a is related to b

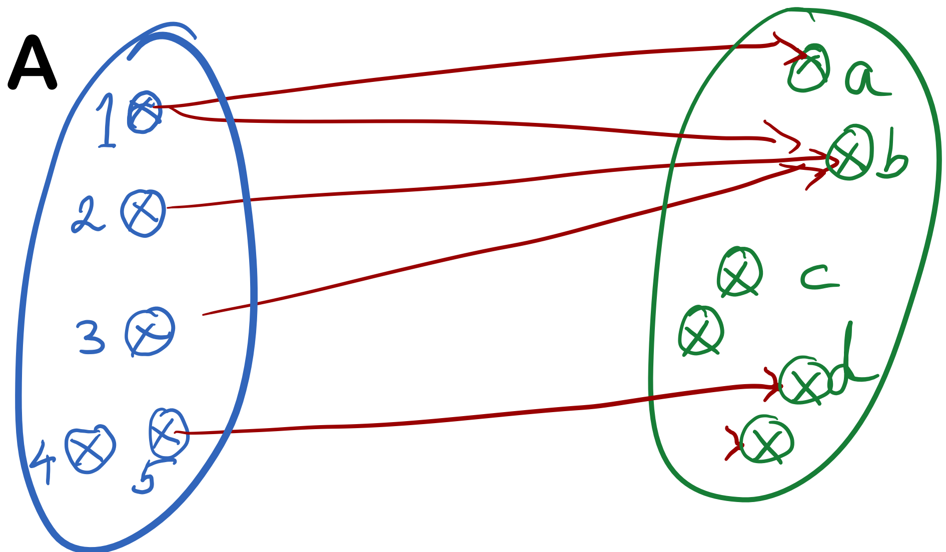


image (2) = {b}
image (1) = {a, b}

preimage (a) = {1}
preimage (b) = {1, 2, 3}
preimg (c) = {}

Range= {a, b, d}

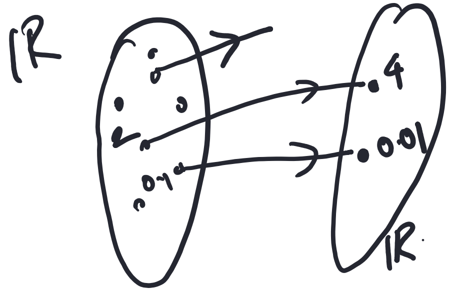
Domain=A={1,2,3,4,5}

Co-domain=B= {a, b, c, d}

Domain Range

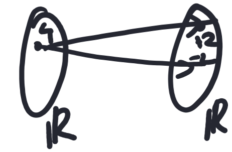
$f: \mathbb{R} \to \mathbb{R}$

$f(x) = x^2$



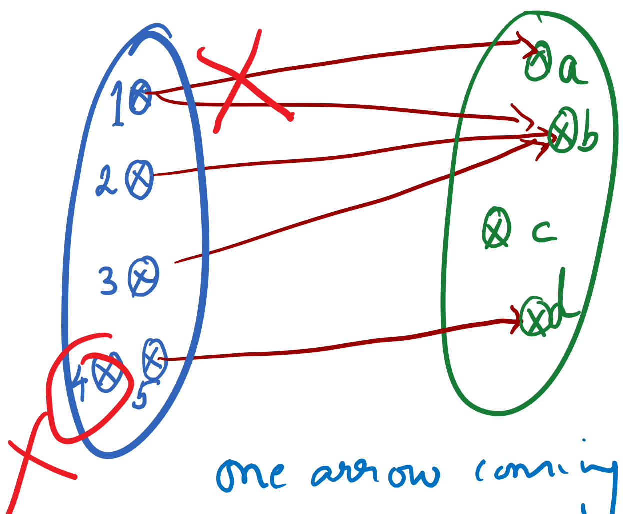
$f(x) = \cancel{x^{1/2}}$

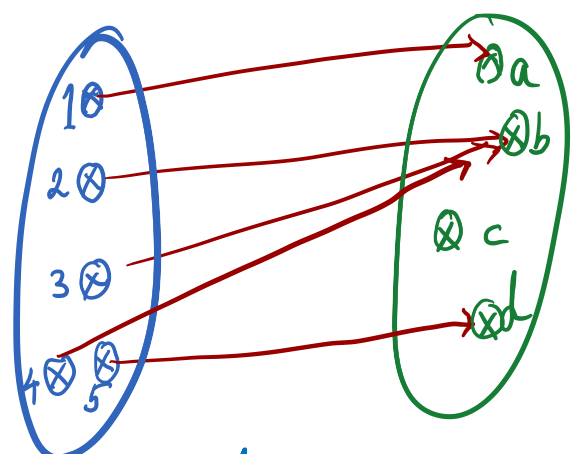
not a function



Maps/ Functions

Function is a binary relation where *every* element of domain is related to exactly one element of the codomain.





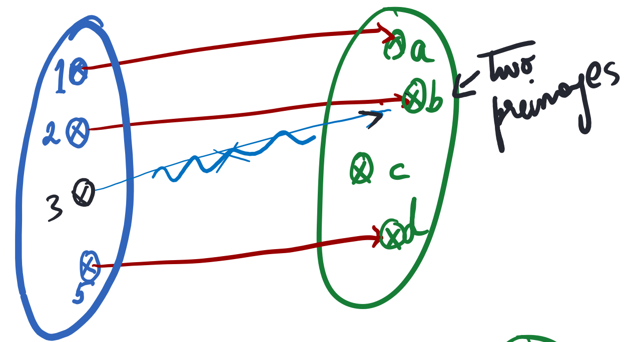
one arrow coming out of every element of the domain

Eg. f: People → Cities representing city of birth

Properties of Functions

- **Injectivity/ One-to-one function:** every element of range has at most one preimage

$|A| \le |B|$

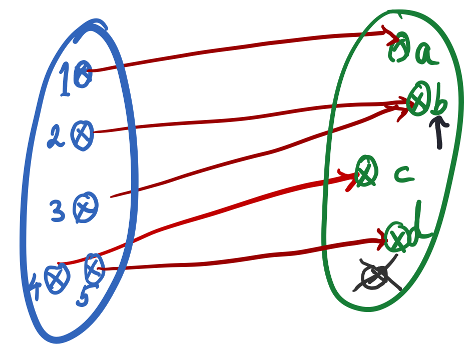


- **Surjectivity/ Onto function:** range=co-domain i.e. every element of co-domain has pre-image.

$|A| \ge |B|$

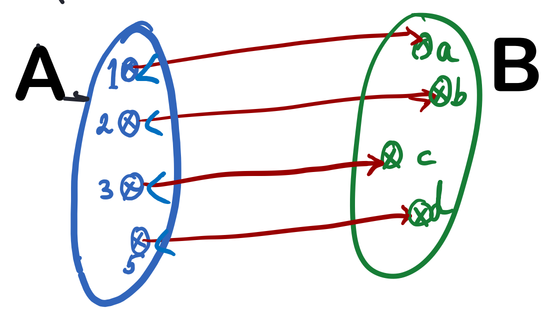
preimage(b) = {2,3}

image(2) = b
image(3) = b



- **Bijective function:** both Injective and Surjective.

$|A| = |B|$



0110

$\{a_2, a_3\}$

Simple Example

- Consider a set $A = \{a_1, a_2, a_3, \dots, a_n\}$. Show a bijective map between the power set of A, and all n-bit strings.

$P(A) \cdot \boxed{2^A}$

$b : P(A) \rightarrow \{0,1\}^n$ n-bit strings

$a_1, a_2, \dots, a_n$

$0/1 \qquad 0,1$

S

Consider a subset $S \subseteq A$.

$\underline{b(S)} \quad b_i(S) = \begin{cases} 1 & \text{if } a_i \in S \\ 0 & \text{if } a_i \notin S \end{cases}$

b is a function since every subset is associated with exactly one n-bit string

(i) Injective: every n-bit string has at most one preimage
(ii) Surjective: every n-bit string at least one preimage

$$\begin{array}{cccc} f: & \mathbb{R} & \to & \mathbb{R}_+ \\ & \cap & & \cap \\ & \mathbb{N} & & \mathbb{N} \end{array}$$

Growth Rate of Functions

Measuring Efficiency, Complexity

How do we measure the efficiency of an algorithm?

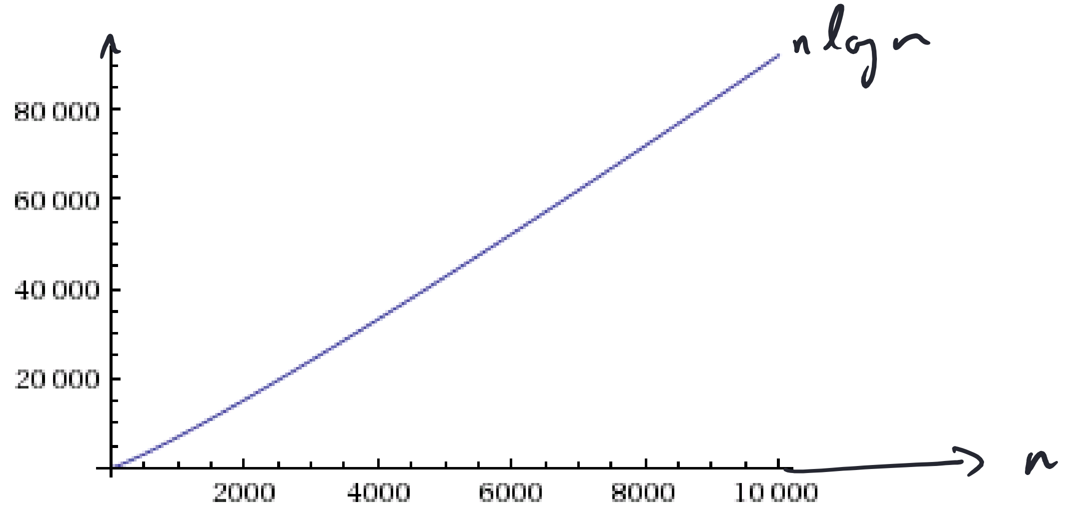
1. Time Complexity = the number of steps taken
2. Space Complexity = the number of memory cells/bits used
3. the amount of energies consumed
4. the size of a VLSI chip
5. the number of gates in a circuit
6. many more ...

Common Functions

Typically, domain and range are numbers (ℝ or ℕ).

Examples: $f(x) = 20x, f(n) = 2n^2 - n, f(n) = e^n$

Sorting numbers:
$f(n) = \text{nlog}_2\, n$

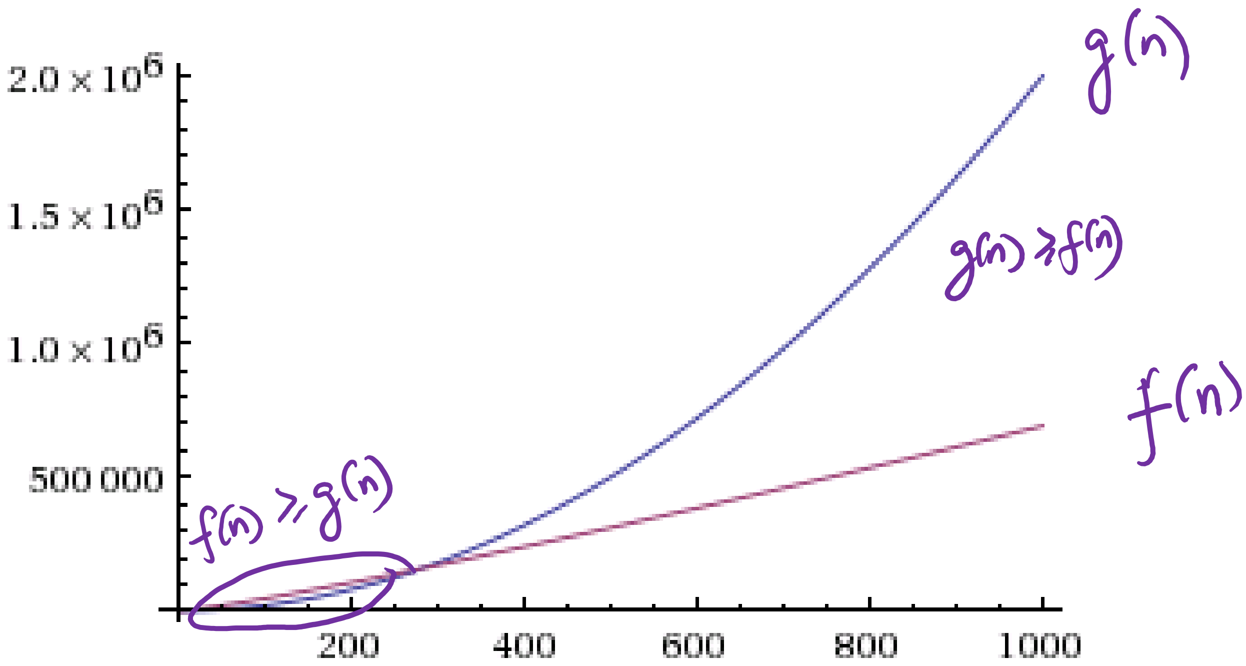


For algorithms: domain is natural algorithms ℕ, range is positive reals $\mathbb{R}^+$

Which is Bigger?

$$f(n) = 100n\log n \quad \text{or} \quad g(n) = 2n^2 - n?$$

$$n = 1,2,3,\ldots?$$

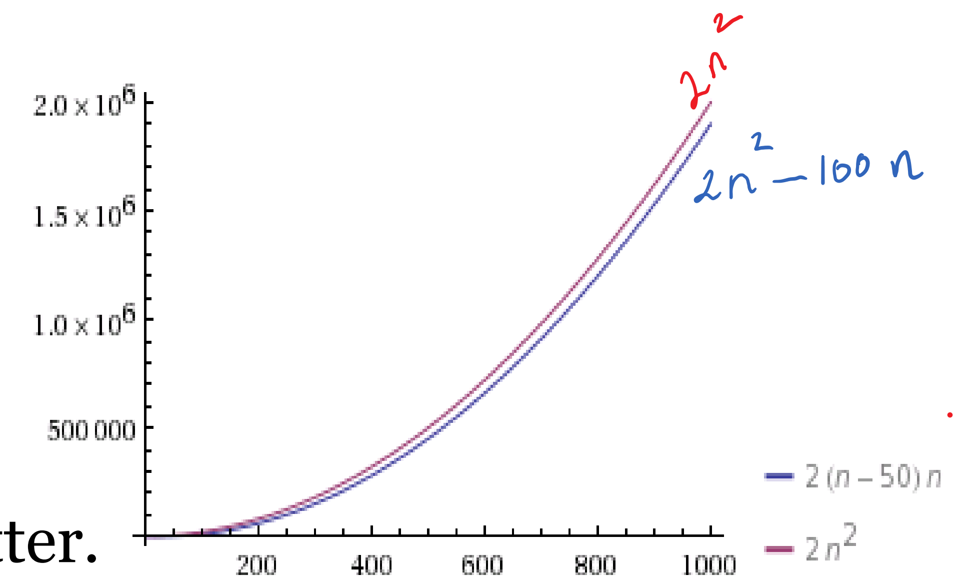


How do we compare?

- Consider large values of n (reason about large inputs)
- Mathematically well-defined and elegant

Asymptotic Analysis:

- Gives good qualitative understanding of function
- Only higher order terms matter.



$g(n) = 2n^2 - 100 \cdot 03n + \sqrt{1001}$
$- \sqrt{1003}\, n \lg n$