CSE 311: Foundations of Computing

Fall 2013

Lecture 9: Set theory and functions

announcements

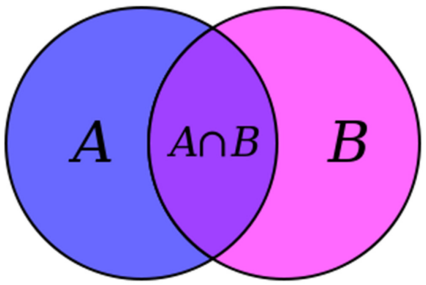
Reading assignment

Set theory

2.1-2.3 (both editions)

set theory

- Formal treatment dates from late 19th century
- Direct ties between set theory and logic
- Important foundational language



definition: a set is an unordered collection of objects

$x \in A$:	"*x* is an element of A" "*x* is a member of A" "*x* is in A"
$x \notin A$:	$\neg (x \in A)$

$A = \{ 1, 2, 7, \text{cat}, \text{dog}, \varphi, \alpha \}$

$\text{cat} \in A$

$\text{fish} \notin A$

definitions

- A and B are *equal* if they have the same elements

$$A = B \equiv \forall x\,(x \in A \leftrightarrow x \in B)$$

- A is a *subset* of B if every element of A is also in B

$$A \subseteq B \equiv \forall x\,(x \in A \rightarrow x \in B)$$

empty set and power set

- Empty set $\emptyset$ does not contain any elements
- Power set of a set A = set of all subsets of A

$$\mathcal{P}(A) = \{\, B : B \subseteq A \,\}$$

cartesian product

$$A \times B = \{\, (a, b) : a \in A, b \in B \,\}$$

set operations

$A \cup B = \{\, x : (x \in A) \vee (x \in B) \,\}$ union

$A \cap B = \{\, x : (x \in A) \wedge (x \in B) \,\}$ intersection

$A - B = \{\, x : (x \in A) \wedge (x \notin B) \,\}$ set difference

$A \oplus B = \{\, x : (x \in A) \oplus (x \in B) \,\}$ symmetric difference

$\overline{A} = \{\, x : x \notin A \,\}$ (with respect to universe U) complement

it's Boolean algebra again

- Definition for $\cup$ based on $\vee$
- Definition for $\cap$ based on $\wedge$
- Complement works like $\neg$

De Morgan's Laws

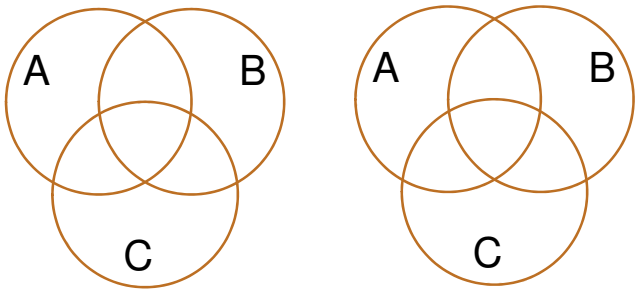
$$\overline{A \cup B} = \bar{A} \cap \bar{B}$$

$$\overline{A \cap B} = \bar{A} \cup \bar{B}$$

Proof technique:
To show C = D show
$x \in C \rightarrow x \in D$ and
$x \in D \rightarrow x \in C$

distributive laws

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$
$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$



representing sets using bits

- Suppose universe U is $\{1,2,\ldots,n\}$
- Can represent set $B \subseteq U$ as a vector of bits:

  $b_1 b_2 \ldots b_n$ where $b_i = 1$ when $i \in B$

  $b_i = 0$ when $i \notin B$

  – Called the *characteristic vector* of set B

- Given characteristic vectors for A and B

  – What is characteristic vector for $A \cup B$? $A \cap B$?

bitwise operations on vectors

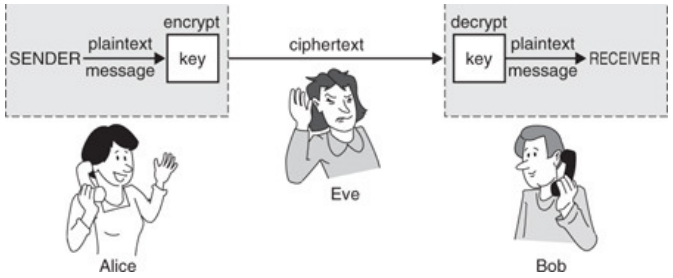
```
  01101101          Java:   z=x|y
∨ 00110111
  01111111

  00101010          Java:   z=x&y
∧ 00001111
  00001010

  01101101          Java:   z=x^y
⊕ 00110111
  01011010
```

a simple identity

- If x and y are bits: $(x \oplus y) \oplus y = ?$
- What if x and y are bit-vectors?

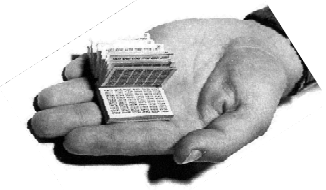
private key cryptography

- **Alice** wants to communicate message secretly to **Bob** so that eavesdropper **Eve** who hears their conversation cannot tell what **Alice**'s message is.
- **Alice** and **Bob** can get together and privately share a secret key K ahead of time.



one-time pad

- Alice and Bob privately share random n-bit vector K
  - Eve does not know K
- Later, Alice has n-bit message m to send to Bob
  - Alice computes $C = m \oplus K$
  - Alice sends C to Bob
  - Bob computes $m = C \oplus K$ which is $(m \oplus K) \oplus K$
- Eve cannot figure out m from C unless she can guess K



unix/linux file permissions

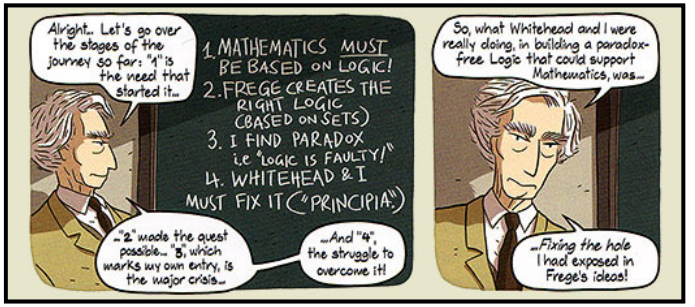
- `ls -l`

```
drwxr-xr-x ... Documents/
-rw-r--r-- ... file1
```

- Permissions maintained as bit vectors
  - Letter means bit is 1
  - "–" means bit is 0.

russell's paradox

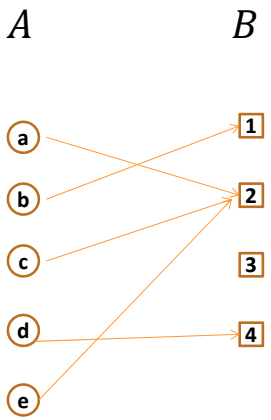
$$S = \{ x : x \notin x \}$$



functions review

- A *function* from A to B
  - an assignment of exactly one element of B to each element of A.
  - We write $f : A \to B$.
  - "Image of a" = $f(a)$
- *Domain* of f: A
- *Range* of f = set of all images of elements of A

image, preimage



is this a function? one-to-one? onto?