Sep 23

Agenda

- Predicates & Quantifiers
- Number Representations

Recall Proposition:

Any statement that is either True or False.

"It is sunny today."

However consider

statement: $x \geq 0$ where x is a real number.

statement: the shape is a square. ; shape can be △, ○, □, ⬠

↳ can be switched out so we don't use 4 diff propositions.

↑ more examples of Predicates

$P(x)$: { True if x has property P

False in all other cases.

Examples: (types of input, property)

1. (Integers, Is it even?) → Even(n)
   ↳ {..., -2, -1, 0, 1, ...}
2. (Natural numbers, Is it prime?) → Prime(n)
   ↳ {0, 1, 2, ...}
3. (Shape, Is it a square?) → Square(S)
4. (pair of integers (d, n); Does d divide n evenly? ↓ no remainder) → Divides(d, n)

Notation:
$d \mid n$
Not required.

Examples of using Predicates

L(n): is leftmost digit of n even? → True: 21, 4, 42
→ False: 12, 3, 34,
→ left-even

E(n): is n even?

F(n): Divides (5, n)

Decent(n): either of 2 conditions
↓
D(n)

i) n is even

ii) n is left-even and 5 divides n.

$$D(n) \equiv E(n) \vee (L(n) \wedge F(n))$$

Side Que: Is this in any normal form?

What if I wanted to test if 20 numbers: $a_1, a_2, a_3, \ldots, a_{20}$ are all decent?

↳ $\equiv D(a_1) \wedge D(a_2) \wedge D(a_3) \wedge \ldots \wedge D(a_{20})$

Quantifiers

① $\forall x$: the predicate that follows is true for all objects under consideration.

"for all x"

eg above: given $a_1, a_2, \ldots, a_{20}$

$$\forall a_i : D(a_i)$$
$$i \leftarrow 1,2,\ldots,20$$

similar to $\wedge$ operation

② $\exists x$: the predicate that follows is True for at least one of the objects being considered.

"there exist x"

eg: given $b_1, b_2, \ldots, b_{10}$

$\exists_{\substack{b_i \\ i \in 1,2,\ldots,10}} : D(b_i) \equiv D(b_1) \vee D(b_2) \vee \ldots \vee D(b_{10})$

similar to $\vee$ operation

③ can also throw a $\neg$ before a quantifier

Translate to Logic

1) Given any integer k, 2k is an even integer.

↳ think of $\forall k$

predicates to use

$Z(k)$: k is an integer

$E(n)$: n is an even integer.

Logic: $\forall k \; Z(k) \Rightarrow E(2k)$

---

2) For any even integer m, there must be another integer k such that $m = 2k$

↳ think ∀

↳ think ∃

$$\forall m \; \exists k: \; Z(m) \wedge E(m) \Rightarrow Z(k) \wedge (m == 2k)$$

Negating Quantifiers

$$\neg \forall x \; P(x)$$

(by def^n) $\equiv \neg \left(P(x_1) \wedge P(x_2) \wedge \cdots \right)$

(DeMorgan) $\equiv \left(\neg P(x_1) \vee \neg P(x_2) \vee \cdots \right)$

(by def^n) $\equiv \left(Q(x_1) \vee Q(x_2) \vee \cdots \right)$

$$\equiv \exists x \; \neg P(x)$$

a) $$\neg \left(\forall x : P(x) \right) \equiv \exists x \; \neg P(x)$$

b) $\neg(\exists x : P(x)) \equiv \forall x \ \neg P(x)$

Number Representations

Binary Computers (at hardware level) can only represent 0s & 1s.

In decimal: $(486)_{10}$ $\quad 0, 1, 2, \ldots, 9$

$= 4 \cdot 100 + 8 \cdot 10^1 + 6 \cdot 1$

$= 4 \cdot 10^2 + 8 \cdot 10^1 + 6 \cdot 10^0$

So in binary: $(10111)_2 = 10111_2$

$= 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$

$= 1 \cdot 16 + 0 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 1 \cdot 1$

$$10111_2 = 23_{10}$$

In Octal: $11_8 = 1 \cdot 8^1 + 1 \cdot 8^0$

$= 1 \cdot 8 + 1 \cdot 1 = 9_{10}$

Other direction from decimal → other base (binary)

$26_{10} \rightarrow ?_2$

26

$= 2 \cdot 13 + 0$

$= 2 \cdot 6 + 1$

$= 2 \cdot 3 + 0$

$= 2 \cdot 1 + 1$

$= 2 \cdot 0 + 1$

↳ stop.

$26_{10} = 11010_2$

Try $26_{10} \rightarrow \underline{\quad ? \quad}_{8}$

$26/8$

$\downarrow$

3 , 2