

27/10/25.

Week-9

- a. Create a Knowledge base consisting of first Order logic statements and prove the given query using Resolution

Proof by Resolution

Given by People KB (a) Premises.

- * John likes all kind of food
- * Apple and vegetables are food
- * Anything anyone eats & not killed is food
- * Anil eats peanuts & still alive
- * Harry eats everything that Anil eats
- * Anyone who is alive implies not killed
- * Anyone who is not killed implies alive.

prove by Resolution that.:

John likes peanut.

Representation in FOL

- a) $\forall x: \text{food}(x) \rightarrow \text{likes}(\text{John}, x)$
- b) $\text{food}(\text{Apple}) \wedge \text{food}(\text{Vegetables})$
- c) $\forall x \forall y: \text{eats}(x, y) \wedge \neg \text{killed}(x) \rightarrow \text{food}(y)$
- d) $\text{eats}(\text{Anil}, \text{peanuts}) \wedge \text{alive}(\text{Anil})$
- e) $\forall x: \text{eats}(\text{Anil}, x) \rightarrow \text{eats}(\text{Harry}, x)$
- f) $\forall x: \neg \text{killed}(x) \rightarrow \text{alive}(x)$
- g) $\forall x: \text{alive}(x) \rightarrow \neg \text{killed}(x)$
- h) $\text{likes}(\text{John}, \text{Peanuts})$

Eliminate implication

- a) $\forall x: \neg \text{food}(x) \vee \text{likes}(\text{John}, x)$
- b) $\text{food}(\text{Apple}) \wedge \text{food}(\text{Vegetables})$
- c) $\forall x \forall y \rightarrow [\text{eats}(x, y) \wedge \neg \text{killed}(x)] \vee \text{food}(y)$
- d) $\text{eats}(\text{Anil}, \text{Peanuts}) \wedge \text{alive}(\text{Anil})$

e). $\forall x \neg \text{eats}(\text{Anil}, x) \vee \text{eats}(\text{Harry}, x)$

f). $\forall x \neg [\neg \text{killed}(x)] \vee \text{alive}(x)$

g). $\forall x \neg \text{alive}(x) \vee \neg \text{killed}(x)$

h). likes (John, Peanuts)

Move negation (\neg) inwards & rewrite

a). $\forall x \neg \text{food}(x) \vee \text{likes}(\text{John}, x)$

b). food (Apple) \wedge food (Vegetables)

c). $\forall x \forall y \neg \text{eats}(x, y) \vee \text{killed}(x) \vee \text{food}(y)$

d). eats (Anil, Peanuts) \wedge alive (Anil)

e). $\forall x \neg \text{eats}(\text{Anil}, x) \vee \text{eats}(\text{Harry}, x)$

f). $\forall x [\text{killed}(x)] \vee \text{alive}(x)$

g). $\forall x \neg \text{alive}(x) \vee \neg \text{killed}(x)$

h). likes (John, Peanuts)

Rename variable @ Standardize variables

a). $\forall x \neg \text{food}(x) \vee \text{likes}(\text{John}, x)$

b). food (Apple) \wedge food (Vegetables)

c). $\forall y \forall z \neg \text{eats}(y, z) \vee \text{killed}(y) \vee \text{food}(z)$

d). eats (Anil, peanuts) \wedge alive (Anil)

e). $\forall w \neg \text{eats}(\text{Anil}, w) \vee \text{eats}(\text{Harry}, w)$

f). $\forall g [\text{killed}(g)] \vee \text{alive}(g)$

g). $\forall k \neg \text{alive}(k) \vee \neg \text{killed}(k)$

h). likes (John, Peanuts)

Drop universe

a). $\neg \text{food}(x) \vee \text{likes}(\text{John}, x)$

b). food (Apple)

c). food (Vegetables)

d). $\neg \text{eats}(y, z) \vee \text{killed}(y) \vee \text{food}(z)$

e). eat (Anil, Peanuts)

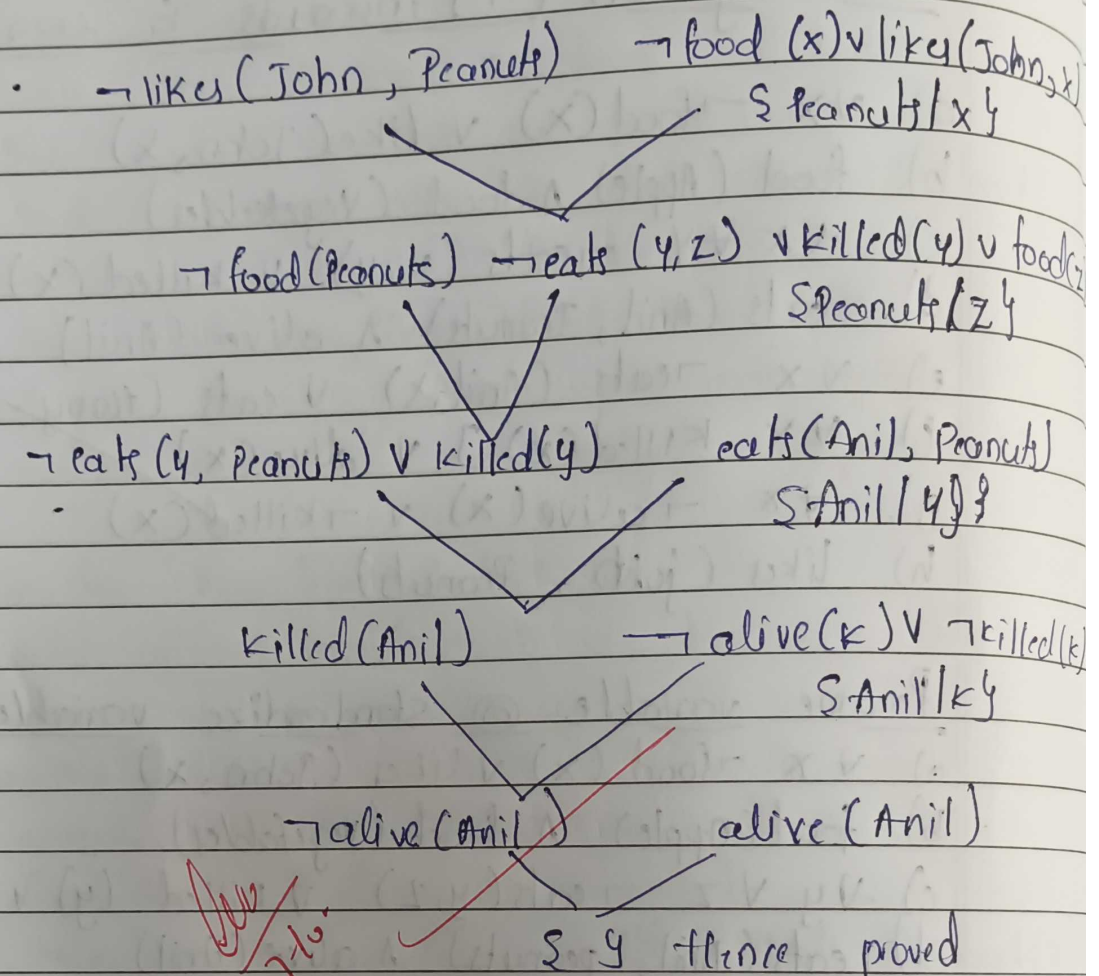
f). alive (Ani)

g). $\neg \text{eats}(\text{Anil}, w) \vee \text{eats}(\text{Harry}, w)$

h). $\text{killed}(g) \vee \text{alive}(g)$

i). $\neg \text{alive}(k) \vee \neg \text{killed}(k)$

j). likes (John, Peanuts).



Algorithm

Algorithm (1). Input: 1. Knowledge Base (KB)
2. Query (Q)

(2) Convert KB & $\neg Q$ to clausal form:

- Eliminate implications
- Move negations inward
- Standardize variables
- Skolemize (remove \exists quantifiers)
- Drop universal quantifiers
- Convert to CNF.

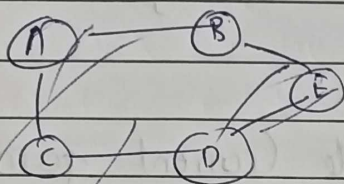
27/10

Week 10

Bafna Gold

Date:

Page:

~~Link~~

A

3) Apply resolution:

- Repeatedly resolve pairs of clauses that contain complementary literals
- Add new clauses to the KB
- Stop if:
 - Empty clause (1) is derived $\rightarrow Q$ is true
 - No new clauses can be added $\rightarrow Q$ is false

4). Output True/False for query Q .

True