Translating English to FOL

- Deb is not tall.

Translating English to FOL

- Every gardener likes the sun.

Translating English to FOL

- You can fool some of the people all of the time.

Translating English to FOL

- You can fool all of the people some of the time.

Translating English to FOL

- All purple mushrooms are poisonous.

Translating English to FOL

- No purple mushroom is poisonous.

Translating English to FOL

- There are exactly two purple mushrooms.

Translating English to FOL

- X is above Y, if X is directly on top of Y, or else there is a pile of one or more other objects directly on top of one another starting with X and ending with Y.

Does Ziggy eat fish?

Generalized Modus Ponens with Horn Clauses

Forward Chaining:

1. $(\forall x)\ cat(x) \Rightarrow likes\ (x, Fish)$
2. $(\forall x)\ (\forall y)\ (cat(x) \wedge likes(x,y) \Rightarrow eats(x,y)$
3. cat(Ziggy)
4. (1), (3) -> likes(Ziggy, Fish)
5. (3), (4), (2) -> eats(Ziggy, Fish)

Generalized Modus Ponens with Horn Clauses

Backward Chaining:

1. $(\forall x)\ cat(x) \Rightarrow likes\ (x, Fish)$
2. $(\forall x)\ (\forall y)\ (cat(x) \wedge likes(x,y) \Rightarrow eats(x,y)$
3. $cat(Ziggy)$

- Goal: eats(Ziggy, Fish) – (2) has eats(x,y) so show:
    - cat(Ziggy) and likes(Ziggy, Fish)

Generalized Modus Ponens with Horn Clauses

- cat(Ziggy) – axiom (3) – 'solved'
- likes(Ziggy, Fish) – (1) has likes(x, Fish) so show
  - cat(Ziggy)
- cat(Ziggy) – axiom (3) again – 'solved'

Rules for Converting FOL wffs to clauses

1. Eliminate $\Leftrightarrow$; replace

   $P \Leftrightarrow Q$ with $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$

2. Eliminate $\Rightarrow$; replace

   $P \Rightarrow Q$ with $\neg P \vee Q$

3. Reduce the scope of $\neg$; replace

   $\neg\neg P$ with P

   $\neg(P \vee Q)$ with $\neg P \wedge \neg Q$

   $\neg(P \wedge Q)$ with $\neg P \vee \neg Q$

   $\neg\forall x P$ with $\exists x \neg P$

   $\neg\exists x P$ with $\forall x \neg P$

Rules for Converting FOL wffs to clauses

4. Standardize Variables; give each quantified variable its own unique name

   eg. $\forall x(P(x) \vee (\exists Q(x))$ with $\forall x P(x) \vee (\exists y Q(y)$

5. Eliminate Existential Quantifiers
6. Eliminate Universal Quantifiers
7. Distribute $\wedge$ over $\vee$; replace

   $(P \wedge Q) \vee R$ with $(P \vee R) \wedge (Q \vee R)$

   $(P \vee Q) \vee R$ with $(P \vee Q \vee R)$

Rules for Converting FOL wffs to clauses

8. Create separate clauses; replace
   $(P(x) \wedge Q(x))$ with $\{P(x), Q(x)\}$
9. Standardize variables apart again so that each clause contains variables names that do no occur in any other clause;

Converting to CNF

- $(\forall x)\ (P(x) \Rightarrow ((\forall y)\ (P(y) \Rightarrow P(f(x,y))) \wedge \neg(\forall\ y)\ (Q(x,y) \Rightarrow P(y))))$

Converting to CNF

1) Eliminate $\Leftrightarrow$; replace $P \Leftrightarrow Q$ with $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$

- $(\forall x)\ (P(x) \Rightarrow ((\forall y)\ (P(y) \Rightarrow P(f(x,y))) \wedge \neg(\forall\ y)\ (Q(x,y) \Rightarrow P(y))))$

Converting to CNF

2) Eliminate $\Rightarrow$; replace
$P \Rightarrow Q$ with $\neg P \vee Q$

- $(\forall x)\ (P(x) \Rightarrow ((\forall y)\ (P(y) \Rightarrow P(f(x,y))) \wedge \neg(\forall y)\ (Q(x,y) \Rightarrow P(y))))$

2

- $(\forall x)\ (\neg P(x) \vee ((\forall y)\ (\neg P(y) \vee P(f(x,y))) \wedge \neg(\forall y)\ (\neg Q(x,y) \vee P(y))))$

Converting to CNF

3) Reduce the scope of ¬;

- $(\forall x)\ (\neg P(x) \vee ((\forall y)\ (\neg P(y) \vee P(f(x,y))) \wedge \neg(\forall y)\ (\neg Q(x,y) \vee P(y))))$

3

- $(\forall x)\ (\neg P(x) \vee ((\forall y)\ (\neg P(y) \vee P(f(x,y))) \wedge (\exists y)\ (Q(x,y) \vee \neg P(y))))$

Converting to CNF

4) Standardize Variables

- $(\forall x)\,(\neg P(x) \vee ((\forall y)\,(\neg P(y) \vee P(f(x,y))) \wedge (\exists y)\,(Q(x,y) \vee \neg P(y))))$

4

- $(\forall x)\,(\neg P(x) \vee ((\forall y)\,(\neg P(y) \vee P(f(x,y))) \wedge (\exists z)\,(Q(x,z) \vee \neg P(z))))$

Converting to CNF

5) Eliminate Existential Quantifiers

- $(\forall x)\ (\neg P(x) \vee ((\forall y)\ (\neg P(y) \vee P(f(x,y))) \wedge (\exists z)\ (Q(x,z) \vee \neg P(z))))$

5

- $(\forall x)\ (\neg P(x) \vee ((\forall y)\ (\neg P(y) \vee P(f(x,y))) \wedge (Q(x,g(x)) \vee \neg P(g(x)))))$

Converting to CNF

6) Eliminate Universal Quantifiers

- $(\forall x)\ (\neg P(x) \lor ((\forall y)\ (\neg P(y) \lor P(f(x,y))) \land (Q(x,g(x)) \lor \neg P(g(x)))))$

6

- $(\neg P(x) \lor ((\neg P(y) \lor P(f(x,y))) \land (Q(x,g(x)) \lor \neg P(g(x)))))$

Converting to CNF

7) Distribute ∧ over ∨

- $(\neg P(x) \vee ((\neg P(y) \vee P(f(x,y))) \wedge (Q(x,g(x)) \vee \neg P(g(x)))))$

7

- $(\neg P(x) \vee \neg P(y) \vee P(f(x,y))) \wedge (\neg P(x) \vee Q(x,g(x))) \wedge (\neg P(x) \vee \neg P(g(x)))$

Converting to CNF

8) Create separate clauses

- $(\neg P(x) \vee \neg P(y) \vee P(f(x,y))) \wedge (\neg P(x) \vee Q(x,g(x))) \wedge (\neg P(x) \vee \neg P(g(x)))$

8

- $\neg P(x) \vee \neg P(y) \vee P(f(x,y))$
- $\neg P(x) \vee Q(x,g(x))$
- $\neg P(x) \vee \neg P(g(x))$

Converting to CNF

9) Standardize variables

- $\neg P(x) \vee \neg P(y) \vee P(f(x,y))$
- $\neg P(x) \vee Q(x,g(x))$
- $\neg P(x) \vee \neg P(g(x))$

9

- $\neg P(x) \vee \neg P(y) \vee P(f(x,y))$
- $\neg P(z) \vee Q(x,g(z))$
- $\neg P(w) \vee \neg P(g(w))$

Mountain People!

- Tom, Bob and Nancy are all members of the Alpine Club of Canada. Every member of the Alpine Club is either a skier or a climber or both. No climber likes rain and all skiers like snow. Nancy dislikes whatever Tom likes and likes whatever Tom dislikes. Tom likes rain and snow.
- Is there a member of the AAC who is a climber but not a skier.

Mountain People - Predicates

- Skier(x) – x is a skier, the domain of x is ACC members
- Climber(x) – x is a climber, the domain of x is ACC members
- Likes(x,y) – x likes y, the domain of x is AAC members and the domain of y is {Rain, Snow}

Mountain People - WFFs

1. $\forall x\ Skier(x) \vee Climber(x)$
2. $\neg\exists x\ Climber(x) \wedge Likes(x, Rain)$
3. $\forall x\ Skier(x) \Rightarrow Likes(x, Snow)$
4. $\forall y\ Likes(Nancy, y) \Leftrightarrow \neg Likes(Tom, y)$
5. $Likes(Tom, Rain) \wedge Likes(Tom, Snow)$
6. $\exists x\ Climber(x) \wedge \neg Skier(x)$ // This is what we want to know.

Mountain People - Clauses

1. $Skier(x_1) \vee Climber(x_1)$
2. $\neg Climber(x_2) \vee \neg Likes(x_2, Rain)$
3. $\neg Skier(x_3) \vee Likes(x_3, Snow)$
4. $\neg Likes(Tom, x_4) \vee \neg Likes(Nancy, x_4)$
5. $Likes(Tom, x_5) \vee Likes(Nancy, x_5)$
6. $Likes(Tom, Rain)$
7. $Likes(Tom, Snow)$
8. $\neg Climber(x_6) \vee Skier(x_6)$

Mountain People Resolution

- 1) $Skier(x_1) \vee Climber(x_1)$ and
  8) $\neg Climber(x_6) \vee Skier(x_6)$ produces:
  - 9)$Skier(x_1)$ $\Theta = \{x_6/x_1\}$
- 9) $Skier(x_1)$ and
  3) $\neg Skier(x_3) \vee Likes(x_3, Snow)$ produces:
  - 10) $Likes(x_1, Snow)$ $\Theta = \{x_3/x_1\}$

Mountain People Resolution

- 10) Likes(x_1, Snow) and
  4) ¬Likes(Tom, x_4) ∨ ¬Likes(Nancy, x_4) produces:
  - 11) ¬Likes(Tom, Snow) Θ = {x4/Snow, x_1/Nancy}
- 11) ¬Likes(Tom, Snow) and
  7) Likes(Tom, Snow) produces
  - 12) □