

create a list of skolem constants

import re:

def fol-to-cnf(fol):

statement = fol.replace("=>", "-")

while '-' in statement:

i = statement.index('-')

new-statement = '[' + statement[:i] + '=>' +
statement[i+1:] + ']' & '[' +
statement[i+1:] + '=>' +
statement[:i] + ']'

statement = statement.replace("=>", "-")

expr = '\[[([^\]]+)\]\]'

statements = re.findall(expr, statement)

for i, s in enumerate(statements):

if '[' in s and ']' not in s:

statements[i] += ']'

for s in statements:

statement = statement.replace(s, fol-to-cnf(s))

while '-' in statement:

~~statement = statement.replace(s,~~

i = statement.index('-')

br = -1 if '[' in statement

new-statement = '~' + statement[br:i] + ']' + statement[i+1:]

statement = statement[:br] + new-statement + statement[br+1:] if br > 0 else
new-statement

while '~' in statement:

i = statement.index('~')

statement = list(statement)

statement[i], statement[i+1], statement[i+2] = ']', statement[i+2], '~'

statement = ''.join(s)

statement = statement.replace('~[A]', '~A')

statement = statement.replace('~[E]', '~E')

expr = '\([A|E]\)'

statements = re.findall(expr, statement)

✱

```

for s in statements:
    statement = statement.replace(s, fol-to-cnf(s))
expr = '~|[[^]]+|'
statements = re.findall(expr, statement)
for s in statements:
    statement = statement.replace(s, DeMorgan(s))
return statement

```

```

print(skolemization(fol-to-cnf("animal(y) <=> loves(x,y)"))
----- ("forall x [forall y [animal(y) => loves(x,y)
=> [exists z [loves(z,x)]]')])
----- ("[american(x) & weapon(y) &
sells(x,y,z) & hostile(z)] =>
criminal(x)")

```

Output:

$[\neg \text{animal}(y) \mid \text{loves}(x,y)] \& [\neg \text{loves}(x,y) \mid \text{animal}(y)]$
 $[\text{animal}(G(x)) \& \neg \text{loves}(x, G(x))] \mid [\text{loves}(F(x), x)]$
 $[\neg \text{american}(x) \mid \neg \text{weapon}(y) \mid \neg \text{sells}(x,y,z) \mid \neg \text{hostile}(z)] \mid$
 ~~$\neg \text{hostile}(z)$~~ $\text{criminal}(x).$

Explanation

$\forall x \text{ King}(x) \wedge \text{greedy}(x) \Rightarrow \text{Evil}(x)$

$\text{King}(\text{Richard}) \wedge \text{greedy}(\text{Richard}) \Rightarrow \text{Evil}(\text{Richard}).$

$A \Leftrightarrow B$
 \hookrightarrow Replace with $(A \Rightarrow B) \wedge (B \Rightarrow A)$
 $A \Rightarrow B$
 $\hookrightarrow \neg(A \wedge \neg B) \sim A \vee B$
 $\neg[A] \rightarrow \text{DeMorgan}$

$\sim [\text{King}(\text{Richard}) \wedge \text{greedy}(\text{Richard})] \vee \text{Evil}(\text{Richard})$
 $\sim \text{King}(\text{Richard}) \vee \neg \text{greedy}(\text{Richard}) \vee \text{Evil}(\text{Richard})$

Sul
 19/1/24

```
39 print(fol_to_cnf("bird(x)=>~fly(x)"))
40 print(fol_to_cnf("∃x[bird(x)=>~fly(x)]"))
```

PROBLEMS



OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
PS C:\Users\neha2\OneDrive\Documents\NehaKamath_1BM21CS113_AILab> python
~bird(x)|~fly(x)
[~bird(A)|~fly(A)]
```

