Lab - 10

Date 11/12/2024
Page

Convert FOL to CNF.

Input first_order_logic statement
Eliminate implication: Replace (A→B) with (¬A∨B)
move ¬(statement) inwards using Demorgan law
Standardize variables: provide each quantifier with unique variables
move quantifiers to the front (prenex form)
Skolemize: Eliminate existential quantifiers by introducing skolem function
Drop universal quantifiers
Distribute ∨ over ∧ to obtain CNF form
Output CNF clauses

output
FOL statement: (A & B) → C
CNF form: ¬A | ¬B | C

cxjnakskl

December 21, 2024

[5]:
```
from sympy import symbols, Or, Not

print("Name: Sudarshan Komar")
print("USN: 1BM22CS291")

# Define variables
x, y = symbols('x y')
P = symbols('P(x, y)')
Q = symbols('Q(x)')
R = symbols('R(y)')

# Step 1: Eliminate Implications
# P(x, y) → (Q(x)  R(y)) is equivalent to ¬P(x, y)  (Q(x)  R(y))
formula_no_implication = Or(Not(P), Or(Q, R))

# Step 2: Skolemization
Skolem_y = symbols('a')  # Skolem constant for y
formula_skolemized = formula_no_implication.subs({y: Skolem_y})

# Final CNF
final_cnf = formula_skolemized

# Display before and after CNF
print("\nBefore CNF:")
print(" x  y (P(x, y) → (Q(x)  R(y)))")

print("\nAfter CNF:")
print(final_cnf)
```

```
Name: Sudarshan Komar
USN: 1BM22CS291

Before CNF:
 x  y (P(x, y) → (Q(x)  R(y)))

After CNF:
Q(x) | R(y) | ~(P(x, y))
```