28/12/20 AI LAB CIE-2 Abhiram G
1BM18CS127

FOL to CNF:

```
def getAttributes(string):
expr = '\([^)]+\)'
matches = re.findall(expr, string)
return [m for m in str(matches) if m.isalpha()]

def getPredicates(string):
 expr = '[a-z ~]+ \([A-Za-z,]+\)'
 return re.findall(expr, string)

def DeMorgan(sentence):
String = ''.join(list(sentence).copy())
String = string.replace('~~', '')
~~stringes~~ flag = '[' in string
string = string.replace('~[', '')
string = string.strip(']')
for predicate in getPredicates(string):
    string = string.replace(predicate, f'~{predicate}')
s = list(string)
for i, c in enumerate(string):
   if c == '|':
      s[i] = '&'
   elif c == '&':
      s[i] = '|'
string = ''.join(s)
string = string.replace('~~', '')
return f'[{string}]' if flag else string
```

p.T.O

Abhiram

```
def skolemization (statemen):
    SKOLEM_CONSTANTS = [f'{chr(c)}' for c in range(
                  ord('A', ord('Z')+1)]
matches = re.findall('[∃].', statement)
for match in matches [::-1]:
    statement = statement.replace(match, '')
    for predicate in getPredicates (statement):
        attributes = getAttributes (predicate)
        if ''.join(attributes).islower():
            statement = statement.replace (match[1],
                    SKOLEM_CONSTANTS.pop(0))
    return statement

import re
def fol_to_cnf (fol):
    statement = fol.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:
        statements = statement.replace (s, fol_to_cnf(s))
    while '-' in statement:
        i = statement.index ('-')
        br = statement.index ('[') if '[' in statement
                                 else 0
        new_statement = '~' + statement [br : i] + '|' + statement [i+1
        statement = statement [:br] + new_statement if
             br > 0 else new_statement
    return skolemization (statement)

Print(fol_to_cnf("∀x (likes(Ram, x) => likes (sita, x)"))
```

Abhiram