## Program:

Convert the following english sentence into FOL. Then convert FOL statement into CNF and use input as All gardiners like the Sun.

Character M. This has been to be a first of

## Algorithm:

if it is the state of the state of A. Bryantote & Environdale def get Attrebuter (string): Bip1 = 1CC133+1) matches = re-fendall (expr., string) return [m for m in str (malches) if m. isalpha (1)

dy getpredicater (string) upl = [a-z~]+\((tA-za-3.]+\) seten re. findall (expr. string) des Derrogan (sentence) sting = '. join (lest (sentence). copy(1) string = 'C' in string string = string. replace (!~ ['. ') string = string. strip ("]'

for pudicate in getpredicate (string): sting = sting. replace (predicate, f'~lpredicate;)

foi i c'in enumente (sting): of c== 'v': ely (== '1': 1, 1, 1, 1)

```
76: 7 = , 11,
                                                  Kevanth. NM
     string = ". join (8)
                                                   18W18C2081
     string = string. replace ("~~", ")
    return f'it strings]' if fleg else string
                                                       Property !
dej skolemization (sentince):
      SKOLEM -CONSTANTS = [f. 'Chaccob']
      statement =". join (list (sentence). copy())
       matches: re. pendall (.'[VII.', statement)
      for match in matches [::-i]:
                                                    Franking/
           Statement = statement. replace (match. ")
           staturent = re. fendall ('(['(" 1]+1]]', staturent)
           for s in statements:
                 Malment = statement. replea (3,8[1:-1])
           for perdicate in getpudicates (statement):
                attributer = getathibuter (pudicales)
                if ". join (attlibuter). i slowers:
                statement = statement, replace (match (1), Soklith. constant.
                  al=[a for a in athibutu if a. islower()]
                  au=[a for a in attributer ig not a.islowers[o]
                  staturent = staturent. replace (au, f'1 SOKLEM-CONSTANTS.
                    c tallos y len (al) else malch [1]3)')
         return statiment.
  dy fol. to-cnf(fol):
        statement: fol. replace ("<=>","-")
        while '-' in statement:
             i = statement. Ender ('-')
                                                     Revanth. NM
```

new-statement: '['+ statement [:i]+'=>'+ statement [+1:] + 'J^C' + statement (9+1:3 +'=>'+ statement

Statement = new - statement statement = statement replace (" => ", "=") expr = "\[([1]]+ )\] statements - re. jendall (cept, shotement) for i, s in enumarale (statements): of 'l' in s and 'J' not in g: Statument [1] += 'J' 2 Burnish &

for & in statements:

11/1/1/18

statement = statement. replace (s. fol-to-cry18) voluile '- ' en statement : i = stationent. Endon(1\_1) be = statement. index (' [') if 'c' in statement else o newstaturent = ' ~ ' + staturent (bisi) + 'v' + staturent (i+): stationent = stationent (: bi) + new\_stationent.

while '~ Y' in statement: i = state muit. induc ('~ 4') staturent = list (staturent) statement [i], statement [i+1], statement [i+2]= ']', statument[i+2] . '~' Statement = ". joen (statement) itumitate ni 'En' shison ; = statement . index (~7')

y bl>0 else new-statement

S= list (statement) Kevanth.NM S(i), 8(i+1), S(i+2)=1A, 'S(i+5)'r~, 1BM18C5081 statiment = '1. join(s) statement = stateprent. replace ("~[4", "[~4") statinuits = re. findall (orpe, statiment) fol s in elatements: Statement & Statement. replace (S. fol-to-cry(S)) 15/ 27/~ 1 = 1920 statements = re-jendall (expr., statement) i in statements: Statement = statement. replace (s. Demorgan(s)) I alway Mills for return statement. 3 townstalet us I I have for -10 - cet (200) From bournels to a place to be to be to be the boundered and the In Diprobable hill I find to the wind to the second to the second to Musical of Comme Lat I to into lake to the market Burneth la more esta artilla the theory tells as your delay Charlana Functions () in the little is the survey to be "E' - Keit Minings. W. Titil Devalue W. O. F. Warrish - Letter to la

Districts of Education of the last of the