M. K. Grogan Resham
18M18CS049
5th CB1

1 Unification -

import re

def getAttributes (expression):

expression = expression.split ("(")["]

expression = "("-join(expression)

expression = expression.split (")")[=-]

expression = ")" join (expression)

attributes = expression.split(",")

ration attributes

deb get Initial Predicate (expression):
return expression split ("(")[0]

del is (enetant (ehan):
return chan is upper () and len (ehan) == 1

del is Variable (char):

return char lower () and lem(char) == 1

def replaceAtttoubutes (exp, ald, new):

attributes = getaltaibutes (exp)
predicate = get Initial Predicate (exp)

for index, val in enumerate (attributes):

16 val = = old:

attributes [ Endex] = new

return predicate + "(" + "," join(attributes)

+ ")"

0

prada

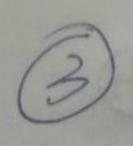
def apply (exp, substitutions): for substitution in substitutions: new, old = substitution exp = replaceAttributes (exp, ald, new) ges menter dol shack occurs (van, exp): if exp. find (var) == -1: return False sure menter getFirstPort (expression): attitutes = get xttbubutes (expression) [0] estudiates number del get Remaining Part (expression): predicate = get Initial Predicate (Expression) attributes = getAltributes (expression) new Expression = prodicate +" ("+", "join( "(" + ([:1] satudiotto neiserapression newter def unify (expl, expa): if exp! == exp 2 return [] if is Constant (expl) and ic Constant (exp2): ef expli-exp2 priot (8" Lexp1 & and Lexp2 & are constants. Commet be unified) return []

if is Constant (exp): notwin [(expl, exp2)] if is constant (exp2): raturn [ (exp2, exp)] is beriable (expz): soturn [(expl, exp2)] if not checkocure (exp1, exp2) else [] if is Variable (exp2): roturn [(exp1, exp2)) if not theckoway (expz, exp1) alse [] if get Initial Bredicate (exp1) ! = get Initial Predi cote (expa): point ("Connet be unified as the predicates de not motch!") sature [7 attribute (ount 1 = len (get Attribute (exp1)) attribute (ount & = len (getAttributes (expa)) if attribute (ount!! = attribute Counts): print (B"length of attailetes fattoribute Count 13 and Eathibute Counts & do not mostoh. Cannot be unified") suturn [] read = get First Part (expl)

read = get First Part (expl)

head a = get First Part (expl)

initial Substitution = unify (head , head a)



if not Initial Substitution: return [] 26 attribute Count 1 = = 1: notwom Initial Substitution tail = got Remaining Part (exp) tail ? = get Remaining Part (exps) if initial Substitution! = []: tail! = apply (tail!, initial Substitution) tail2 = apply(tail2, irrital Substitution) remaining Substitution = unify (tail 1, tail 2) if not samaining Substitution: [] menter rature initial substitution + saméuring Substitution if-name - == "\_main\_": print (" Enter the first pradicate (expression") el=input() prints ("Enter the second expression") ea = input() substitutions = unify (c1, e2) prior ("The substitutions are: ") print (1) o join (substitution) for substitution in substitutions ]