

**Assignment No: 4(B)**

***Prolog- Unification Algorithm***

**Name**: AMIT KUMAR

**Roll No**: 137

**PRN No**: 0120180479

#To check whether predicate is same

def funct\_pred(p):

li = []

for i in p:

if(i == "("):

break

else:

li.append(i)

return li

# To determine number of argument

def funct\_argu(y):

c = 0

for i in y:

if(i == ","):

c += 1

return c

# To convert string into list

def split(s):

countstr = 1

nlen=len(s)

for i in s:

if (i == "("):

d = s[countstr:nlen-1]

countstr +=1

e=d.split(",")

return e

# Function used to determine Variables of both input a & b

def f1(c,d):

global count

global flag1

for z in c:

something=z

for i in range(count,count+1):

if((something[0] >= "A" and something[0] <= "Z") or something[0] == "\_"):

count +=1

arr\_res.append(z)

arr\_res.append(d[i])

flag1 = 1

break

else:

x=f2(c,d)

if x != "NULL":

flag1 = 1

arr\_res.append(x)

arr\_res.append(z)

break

else:

print("False:Variable is not well defined")

flag1 = 0

exit()

def f2(e,f):

global count

global flag1

for w in range(count,count+1):

anything=f[w]

if((anything[0] >= "A" and anything[0] <= "Z") or anything[0] == "\_"):

count +=1

flag1 = 0

return anything

else:

x="NULL"

return x

# To display the output

def display(arr\_dis):

global flag1

arr\_dis.reverse()

len1 = len(arr\_dis)

if flag1 == 1:

for i in range(0,len1,2):

var = arr\_dis.pop(-1)

val = arr\_dis.pop(-1)

print(var, "=", val)

else:

print("False: Variable is not defined")

a = str(input("Enter the L.H.S of expression:"))

b = str(input("Enter the R.H.S of experssion:"))

print(a,"=",b)

flag1 = bool

count = 0

arr\_res = []

if(a==b):

print("True")

else:

pred1=funct\_pred(a)

pred2=funct\_pred(b)

argu1=funct\_argu(a)

argu2=funct\_argu(b)

arr\_a = split(a)

arr\_b = split(b)

if(pred1 == pred2 and argu1 == argu2 and arr\_a == arr\_b):

print("True: The above is unified")

elif(pred1 == pred2 and argu1 == argu2):

f1(arr\_a,arr\_b)

display(arr\_res)

elif(pred1 != pred2):

print("The predicates are not same")

elif(argu1 != argu2):

print("The number of argument are not same")

else:

print("\*\*\*\*\*False\*\*\*\*")

**OUTPUT**:

#Enter the L.H.S of expression:p(Amit,105)

#Enter the R.H.S of experssion:p(102,Ajay)

#p(Sarfaraj,105) = p(102,Ajay)

#Sarfaraj = 102

#Ajay = 105

#Enter the L.H.S of expression:name(x,y)

#Enter the R.H.S of experssion:name(Viraj,Vijay)

#name(x,y) = name(Viraj,Vijay)

#Viraj = x

#Vijay = y

#Enter the L.H.S of expression:p(x,Y,8)

#Enter the R.H.S of experssion:p(P,Q,R)

#p(x,Y,8) = p(P,Q,R)

#P = x

#Y = Q

#R = 8

#Enter the L.H.S of expression:parent

#Enter the R.H.S of experssion:parent

#parent = parent

#True

#Enter the L.H.S of expression:5

#Enter the R.H.S of experssion:5

#5 = 5

#True

#Enter the L.H.S of expression:part(X,Y,Z)

#Enter the R.H.S of experssion:part(X,Y,Z)

#part(X,Y,Z) = part(X,Y,Z)

#True

#Enter the L.H.S of expression:part(a,b,c)

#Enter the R.H.S of experssion:pair(x,y,z)

#part(a,b,c) = pair(x,y,z)

#The predicates are not same

#Enter the L.H.S of expression:name(A,B)

#Enter the R.H.S of experssion:name(Amit)

#name(A,B) = name(Amit)

#The number of argument are not same