Jai Jain
1BM18CS040

AI LAB TEST -2

Q1 Infer whether the given pair of statements can be unified or not. If unification is possible write the code for substitutions.
Justify your answer for these cases also along with successful unification

1) Predicates are different
2) Mismatch in Number of arguments
3) If the arguments are constants

Program for unification:

```
noPred = 0
noArg = [None for i in range(10)]
nouse = ' '
predicate = [None for i in range(10)]
argument = [[None for i in range(10)] for i in range(10)]

def main()
    global noPred
    char = 'y'
    while(char == 'y'):
        print("Program for Unification")
        noPred = int(input("Enter Number of Predicates"))
        for i in range(noPred):
            print("Enter Predicate", (i+1), ":")
            predicate[i] = input()
            print('Enter no. of Arguments for
                  each predicate", predicate[i]:")
```

Jai Jaim
1BM18CS040
AI LAB TEST - 2

Century

Page No.
Date :

```
noArg[i] = int(input())
for j in range(noArg[i]):
    print("Enter argument", j+1, ":")
    argument[i][j] = input()

printPredicate()
check_arg_pred()
char = input("Do you want to continue(y/n):")

def printPredicate():
    print("Predicate's Are :")
    for i in range(noPredicate):
        print(predicate[i], "(", end=" ")
        for j in range(noArg[i]):
            print(argument[i][j], end=" ")
            if(j != noArg[i]-1):
                print(",", end=" ")
        print(")")

def unify():
    flag = 0
    for i in range(noPred-1):
        for j in range(noArg[i]):
            if(argument[i][j] != argument[i+1][j]):
                flag = 0
                if(flag==0):
                    print("Substitution is")
                    print(argument[i+1][j], "/",
                          argument[i][j])
                    flag += 1
```

Jai Jain
1BM18CS040
AI LAB TEST - 2

Century

Page No.
Date :

```
    if (flag == 0):
        print ("Arguments are Identical")
        print ("No need of Substitution")
        flag += 1

def check_arg_pred():
    predflag = 0
    argflag = 0
    for i in range (noPred - 1)
        if (predicate[i] != predicate[i+1])
            print ("Predicates are not same")
            print ("Unification cannot progress")
            predflag = 1
            break
    if (predflag != 1):
        ind = 0
        key = noArg[ind]
        length = len(noArg)
        for i in range (0, key-1):
            if i >= key:
                continue
            if ind != length - 1:
                ind += 1
                key = noArg[ind]
            if (noArg[i] != noArg[i+1]):
                print ("No of arguments are not same")
                argflag = 1
                break
        if (argflag == 0 and predflag != 1):
            unify()
main()
```