



Atma Ram Sanatan Dharma College

Class Assignment

Practical File

Question 2

SUBMITTED BY

Name : Ankit Sarawag
Course : Bsc.(Hons) Computer Science
Roll no : 22/28006
Semester : 2
Subject : Discrete Mathematical Structures
Teacher : Dr. Shalini Gupta(Faculty Of
Computer Science)

2) Create a class Relation, use matrix notation to represent a relation. Include member functions to check if the relation is Reflexive, Symmetric, Anti-Symmetric, Transitive. Using these functions check whether the given relation is Equivalence or Partial Order relation or None.

CODE

```
question2 > 2.py > RELATION > determineRelationType
1 class RELATION: #declaring the class
2     def __init__(self,matrix):
3         self.matrix=matrix
4         self.lenOfMatrix=len(matrix)
5     def isReflexive(self): #function checking reflexivity of the relation
6         for i in range(self.lenOfMatrix):
7             if self.matrix[i][i]!=1:
8                 return False
9         return True
10    def isSymmetric(self): #function checking symmetric nature of the relation
11        for i in range(self.lenOfMatrix):
12            for j in range(self.lenOfMatrix):
13                if self.matrix[i][j]!=self.matrix[j][i]:
14                    return False
15        return True
16    def isAntiSymmetric(self): #function checking antisymmetrix nature of the relation
17        for i in range(self.lenOfMatrix):
18            for j in range(self.lenOfMatrix):
19                if self.matrix[i][j]==1 and self.matrix[j][i]==1 and i!=j:
20                    return False
21        return True
22
23    def isTransitive(self): #function checking transitive nature of the relation
24        for i in range(self.lenOfMatrix):
25            for j in range(self.lenOfMatrix):
26                if self.matrix[i][j]==1:
27                    for k in range(self.lenOfMatrix):
28                        if self.matrix[j][k]==1 and self.matrix[i][k]!=1:
29                            return False
30        return True
31
32    def determineRelationType(self): #function checking relation type
33        isEquivalence=self.isReflexive() and self.isSymmetric() and self.isTransitive()
34        isPartialOrder=self.isReflexive() and self.isAntiSymmetric() and self.isTransitive()
35
36        if isEquivalence:
37            return "The given relation is a Equivalence Relation"
38        elif isPartialOrder:
39            return "The given relation is a partial order Relation"
40        else:
41            return "None"
42
43    matrix1=[[1,0,0],[0,1,0],[0,0,1]] #relation 1
44    Relation1=RELATION(matrix1)
45    print(Relation1.determineRelationType())
46
47    matrix2=[[1,0,0],[0,1,1],[0,0,1]] #relation 2
48    Relation2=RELATION(matrix2)
49    print(Relation2.determineRelationType())
```

Output

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\ankit\Desktop\DMS> python -u "c:\Users\ankit\Desktop\DMS\question2\2.py"
```

```
The given relation is a Equivalence Relation
```

```
The given relation is a partial order Relation
```

```
PS C:\Users\ankit\Desktop\DMS>
```