ReadMe

Programming Language Used: C++
Library Used: Bits/stdc++.h
Logic Used: 1. Standard input is taken in the form of ordered pairs like (1,1), (2,2) etc.
2. Relation is represented in the form of the matrix and a[i][j]=1 if (i,j) belongs to the relation R, else a[i][j]=0.
3. If relation is reflexive (all diagonal elements are 1), antisymmetric (if a[i][j]=1 then a[j][i]=0), and transitive (a[i][j]=1 and a[j][k]=1 then a[i][k] is also 1) then it is a poset.
4. If every 2 elements of poset have LUB and GLB then it is a lattice.
5. For every 2 elements of a poset to have LUB and GLB they must be comparable. If they are incomparable, they can't have LUB and GLB.
How to compile- Copy the given code as it is on any online C++ compiler and run the program and give the inputs. Link for one online compiler is- https://www.onlinegdb.com/online_c++_compiler

OR

Copy the code on any IDE with C++ compiler installed on your computer. Build the program and run.

Code Screen Shot-

```cpp
//Amogh Garg-2020UCO1688
#include<bits/stdc++.h>
using namespace std;

int main(){
    //Entering number of ordered pairs
    int N;
    cout<<"Enter the number of ordered pairs:";
    cin>>N;
    //Creating a 2-D matrix for representing relation
    int arr[4][4]={0}; //Initially all elements of matrix are 0
    //Domain set A={0,1,2,3}
    //Entering the ordered pairs and storing them in matrix
    for(int i=0;i<N;i++){
        int a,b;
        cout<<"Enter the elements of ordered pair:";
        cin>>a;
        cin>>b;
        arr[a][b]=1; //Representing relation in the matrix
    }
    //Checking for Poset
    //1.Reflexivity
    bool isreflexive= true;
    for(int i=0;i<4;i++){
        if(arr[i][i] == 0){ //If (a,a) not in matrix -> Not reflexive
            isreflexive= false;
            break;
        }
```

```
        }
    }
    if(isreflexive){
        cout<<"The relation is Reflexive!"<<endl;
    }
    else{
        cout<<"The relation is NOT Reflexive!"<<endl;
    }
    //2.Anti-Symmetricity
    bool isantisymm= true;
    for(int i=0;i<4;i++){
        for(int j=0;j<4;j++){
            if(i!=j && arr[i][j]==1 && arr[j][i]==1){ //If (a,b) in R then (b,a) in R only if a=b
                isantisymm=false;
                break;
            }
        }
    }
    if(isantisymm){
        cout<<"The relation is AntiSymmetric!"<<endl;
    }
    else{
        cout<<"The realtion is NOT AntiSymmetric!"<<endl;
    }
    //3.Transitivity
    bool istransitive= true;
    for(int i=0;i<4;i++){
        for(int j=0;j<3;j++){
```

```
    for(int i=0;i<4;i++){
        for(int j=0;j<3;j++){
            for(int k=j+1;j<4;j++){
                //If (a,b) in R and (b,c) in R then (a,c) also in R
                if(arr[i][j]==1 && arr[j][k]==1 && arr[i][k] != 1){
                    istransitive= false;
                }
            }
        }
    }
    if(istransitive){
        cout<<"The relation is Transitive!"<<endl;
    }
    else{
        cout<<"The relation is NOT Transitive!"<<endl;
    }
    // If all conditions are true then it is POSET
    if(istransitive && isantisymm && isreflexive){
        cout<<"The set is a POSET!"<<endl;
    }
    //If set is not POSET then it is not a LATTICE also
    else{
        cout<<"The set is neither POSET nor LATTICE!"<<endl;
        //End program
        return 0;
    }
    //Checking for LATTICE-> Every 2 elements must be comparable
    //Checking whether every 2 elements are comparable or not
```

```
    int l[4]={0},g[4]={0};
    for(int i=0;i<4;i++){
        for(int j=0;j<4;j++){
            if(arr[i][j]==1){
                switch(i){
                    case 0:l[0]++;break;
                    case 1:l[1]++;break;
                    case 2:l[2]++;break;
                    case 3:l[3]++;break;
                }
                switch(j){
                    case 0:g[0]++;break;
                    case 1:g[1]++;break;
                    case 2:g[2]++;break;
                    case 3:g[3]++;break;
                }
            }
        }
    }
    //Check condition
    bool x=false,y=false;
    for(int i=0;i<4;i++){
        if(l[i]==4){
            x=true;
        }
        if(g[i]==4){
            y=true;
        }
    }
    //Printing
```

```
            x=true;
        }
        if(g[i]==4){
            y=true;
        }
    }
    //Printing
    if(x && y){ //If x and y both are true->All elements are comparable pairwise
        cout<<"The given poset is LATTICE"<<endl;
    }
    else{
        cout<<"The given poset is NOT A LATTICE!"<<endl;
    }



    return 0;
}
```

By: Amogh Garg
2020UCO1688