**Problem 2 :**

**Problem Statement :**

Alice resides in Wonderland, a nation with N cities ranging in number from 1 to N and forming a tree. In other words, there exist (N-1) undirected routes that connect these N cities, allowing every pair of cities to communicate with one another. The distance between cities i and j is computed as (i+j). Consequently, a brilliant scientist named Trudy in Wonderland created a completely novel car that allows Alice to travel between cities in f(i, j) time, where f(i, j) represents the bitwise XOR of the lengths of the roads along the direct route between cities i and j. Alice was therefore interested to discover the bitwise XOR of all f(i, j) such that i < j. Help Alice to complete his task.

**Input :**

The first line contains N (the number of cities in the Wonderland)

• Then (N-1) lines contain two integers u and v if there is a road between u and v

• The offered roads will undoubtedly form a tree.

**Logic :**

Here , We Have to Calculate Distance between two cities / vertices by using Bit wise XOR operator. For example , distance between 1(001) and 2(010) is 3(011). We have to do with all edges. To calculate final distance, we must do bitwise XOR of all calculated distances.

**Algorithm :**

* Take Input N Cities
* Count routes Cities -1
* Make result array of size routes
* Run a for loop which takes the cites distances and count their bitwise XOR with condition (i<j)
* if some pair does not satisfy Place 0 at that position in result array
* Store that bitwise XOR in an array which was created earlier
* Perform bitwise XOR operation of each element in array
* Print result of the above operation

**Code :**

#include <bits/stdc++.h>

using namespace std;

int main()

{

int a,b,cities,routes;

cin>>cities;

routes=cities-1;

int timearray[routes];

for (size\_t i = 0; i < routes; i++)//Calculation of bitwise XOR operation of two routes i and j

{

cin>> a ;

cin>> b ;

if(a<b)

{

timearray[i]=a^b;

}

else

timearray[i]=0;

}

int res = 0;

for (int i = 0; i < routes; i++)//Calculate of bitwise XOR operation of F(i,j)

{

res ^= timearray[i];

}

cout << res;

return 0;

}

**Output :**

A screen shot of a computer

Description automatically generated