ASSIGNMENT

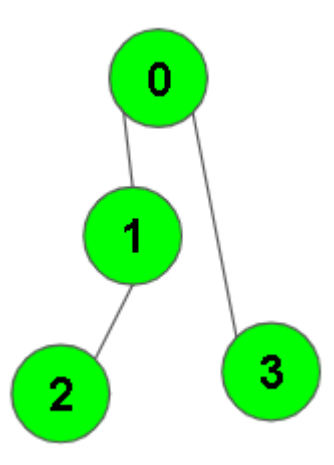
1.Given a tree with **N** nodes rooted at **1**. Each node labeled with a value **arr[i]**. The task is to find the absolute difference between the sum of values of nodes at even level and odd level.

Example:
Input:
4
1 2 3 4
1 2
3 4
4 1

Output:
2

2. Given a connected undirected graph. Perform a Depth First Traversal of the graph.

Input:



Output: 0 1 2 3

3. You are given an array A[] of n-elements. There are two players Alice and Bob. A Player can choose any of element from array and remove it. If

the bitwise XOR of all remaining elements equals 0 after removal of selected element, then that player looses. This problem is variation of nim-game.

Examples :

Input : A[] = {3, 3, 2}
Output : Winner = Bob
Explanation : Alice can select 2 and remove it that make XOR of array equ als to zero also if Alice choose 3 to remove than Bob can choose any of 2/ 3 and finally Alice have to make his steps

4. Given a set of **N** nuts of different sizes and **N** bolts of different sizes. There is a one-one mapping between nuts and bolts. Match nuts and bolts efficiently.

Comparison of a nut to another nut or a bolt to another bolt is not allowed. It means nut can only be compared with bolt and bolt can only be compared with nut to see which one is bigger/smaller.
The elements should follow the following order ! # $ % & * @ ^ ~ .

Input:

N = 5

nuts[] = {@, %, $, #, ^}

bolts[] = {%, @, #, $ ^}

Output:

$ % @ ^

$ % @ ^

5. You are given a pointer/ reference to the node which is to be deleted from the linked list of **N** nodes. The task is to delete the node. Pointer/ reference to head node is not given.
Note: No head reference is given to you. It is guaranteed that the node to be deleted is not a tail node in the linked list.

Input:

N = 4

value[] = {10,20,4,30}

node = 20

Output: 10 4 30

6. A number **n** can be broken into three parts **n/2, n/3 and n/4** (consider only **integer** part). Each number obtained in this process can be divided further recursively. Find the **maximum sum** that can be obtained by summing up the divided parts together.
Note: The maximum sum may be obtained without dividing n also.

Input:

n = 12

Output: 13

7. You are given a special linked list with **N** nodes where each node has a next pointer pointing to its next node. You are also given **M** random pointers, where you will be given **M** number of pairs denoting two nodes **a** and **b** **i.e. a->arb = b.**

Construct a copy of the given list. The copy should consist of exactly **N** new nodes, where each new node has its value set to the value of its corresponding original node. Both the next and random pointer of the new nodes should point to new nodes in the copied list such that the pointers in the original list and copied list represent the same list state. None of the pointers in the new list should point to nodes in the original list.

Input:

N = 4, M = 2

value = {1,2,3,4}

pairs = {{1,2},{2,4}}

Output: 1

8. Given two integers a and b. Find the sum of two numbers **without using arithmetic operators**.

Example 1:

Input:

a = 5, b = 3

Output: 8

9. Given an incomplete Sudoku configuration in terms of a 9x9 2-D square matrix(**mat[][]**) the task to check if the current configuration is valid or not where a 0 represents an empty block.

Example 1:

Input: mat[][] = [

[3, 0, 6, 5, 0, 8, 4, 0, 0]

[5, 2, 0, 0, 0, 0, 0, 0, 0]

[0, 8, 7, 0, 0, 0, 0, 3, 1]

[0, 0, 3, 0, 1, 0, 0, 8, 0]

[9, 0, 0, 8, 6, 3, 0, 0, 5]

[0, 5, 0, 0, 9, 0, 6, 0, 0]

[1, 3, 0, 0, 0, 0, 2, 5, 0]

[0, 0, 0, 0, 0, 0, 0, 7, 4]

[0, 0, 5, 2, 0, 6, 3, 0, 0]

]

Output: 1

Explaination: It is possible to have a

proper sudoku.

10. Aterp is the head nurse at a city hospital. City hospital contains R*C number of wards and the structure of a hospital is in the form of a 2-D matrix.
Given a matrix of dimension **R*C** where each cell in the matrix can have values 0, 1, or 2 which has the following meaning:
0: Empty ward
1: Cells have uninfected patients
2: Cells have infected patients

An infected patient at ward [i,j] can infect other uninfected patient at indexes [i-1,j], [i+1,j], [i,j-1], [i,j+1] (**up**, **down**, **left** and **right**) in unit time. Help Aterp determine the minimum units of time after which there won't remain any uninfected patient i.e all patients would be infected. If all patients are not infected after infinite units of time then simply return -1.

Input:

3 5

2 1 0 2 1

1 0 1 2 1

1 0 0 2 1

Output:

2

Explanation:

Patients at positions {0,0}, {0, 3}, {1, 3}

and {2, 3} will infect patient at {0, 1},

{1, 0},{0, 4}, {1, 2}, {1, 4}, {2, 4} during 1st

unit time. And, during 2nd unit time, patient at

{1, 0} will get infected and will infect patient

at {2, 0}. Hence, total 2 unit of time is

required to infect all patients.