Celebrity Problem

A person is a celebrity if it is known to everyone but himself knows no one.

BRUTE FORCE :

For every person find out the people one knows and number of people who knows the particular person.

If even the current person knows 1 person he cannot be a celebrity or if is not known to even 1 person then it cannot be a celebrity. Make these checks for every person and wherever these condition fail the person is a celebrity else no celebrity exists thus return -1.

Code :

```
#include <bits/stdc++.h>
/*
  This is signature of helper function 'knows'.
  You should not implement it, or speculate about its implementation.

  bool knows(int A, int B);
  Function 'knows(A, B)' will returns "true" if the person having
  id 'A' know the person having id 'B' in the party, "false" otherwise.
*/

int findCelebrity(int n) {
  // Write your code here.
   bool know1=false,know2=false;
  for(int i=0;i<n;i++)
  {
    know1=false,know2=false;
    for(int j=0;j<n;j++)
    {
      if(i!=j)
      {
        know1|=knows(i,j);
        know2|=knows(j,i);
        if(know2==false&&know1==true)
        {
          break;
        }

      }
    }
```

```
    if(know1==false&&know2==true)
      return i;
  }
  return -1;
}
```

- Time Complexity : O(n^2)
- Space Complexity : O(1)

OPTIMAL APPROACH :

Use a stack to push all the person into stack and pop two person at a time and compare whether they know each other is A knows B then A cannot be a celebrity thus push B back into stack, if B knows A then push A back into stack as B cannot be a celebrity. Repeat until the stack size is >1.

If the stack becomes empty no celebrity found and return -1.

If the stack contains 1 element this can be a potential celebrity make a check by just checking whether its full row is 0 except diagonal i.e (k,k) which means it knows no one and whole column is 1 which means everyone knows him.

If this condition is satisfies return this person.

Code :

```
#include <bits/stdc++.h>
/*
  This is signature of helper function 'knows'.
  You should not implement it, or speculate about its implementation.

  bool knows(int A, int B);
  Function 'knows(A, B)' will returns "true" if the person having
  id 'A' know the person having id 'B' in the party, "false" otherwise.
*/

int findCelebrity(int n) {
  // Write your code here.
   stack<int> s;
   for(int i=0;i<n;i++)
   {
     s.push(i);
   }
   while(s.size()>1)
   {
     int A=s.top();
```

```
      s.pop();
      int B=s.top();
      s.pop();
      if(knows(A,B))
      {
        s.push(B);
      }
      else if(knows(B,A))
      {
        s.push(A);
      }
    }
    if(!s.empty())
    {
     int tempAns=s.top();
     int cnt1=0,cnt0=0;
     for(int i=0;i<n;i++)
     {
     if(i!=tempAns)
     {
       if(knows(i,tempAns))
       {
         cnt1++;
       }
       if(!knows(tempAns,i))
       {
         cnt0++;
       }
     }

     }
     if (cnt1 == n - 1 && cnt0 == n - 1) {
       return tempAns;
     }
       }
          return -1;
}
```

- Time Complexity : O(N)
- Space Complexity : O(N)