**UNIQUE NUMBERS - II**

We are given an array containg n numbers. All the numbers are present twice except for two numbers which are present only once. Find the unique numbers in linear time without using any extra space. ( Hint - Use Bitwise )

**Input Format:**

First line contains the number n. Second line contains n space separated number.

**Constraints:**

n < 10^5

**Output Format**

Output a single line containing the unique numbers separated by space

**Sample Input**

4

3 1 2 1

**Sample Output**

2 3

Program-

#include <iostream>

using namespace std;

int main()

{

int a[100005];

int n,i,ans=0,no;

cin>>n;

for(i=0;i<n;i++)

{

cin>>no;

a[i]=no;

ans=ans^no;

}

int temp=ans;

int pos=0;

while((temp&1)!=1)

{

pos++;

temp=temp>>1;

}

//The first set bit in ans is at postion 'pos'

int mask=(1<<pos);

//find those number which contains set bit at position pos

int x=0,y=0;

for(i=0;i<n;i++)

{

if((a[i]&mask)>0)

{

x=x^a[i];

}

}

y=ans^x;

cout<<min(x,y)<<" "<<max(x,y);

}