

```
1 #include<bits/stdc++.h>
2 using namespace std;
3 struct trie
4 {
5     struct trie* child[2];
6     int value;
7 }*root=NULL;
8 void insertvalue(int);
9 struct trie* createnode();
10 int xorfinder(int);
11 int main()
12 {
13     int A[]={7,3,9,12};
14     int B[]={1,3,5,2};
15     int N=sizeof(A)/sizeof(int);
16     root=createnode();
17     for(int i=0;i<N;i++)
18     {
19         insertvalue(B[i]);
20     }
21     for(int i=0;i<N;i++)
22     {
23         cout<<xorfinder(A[i])<<" ";
24     }
25     return 0;
26 }
27 struct trie* createnode()
28 {
29     struct trie* temp=new struct trie;
30     temp->value=0;
31     temp->child[0]=NULL;
32     temp->child[1]=NULL;
33     return temp;
```

```

34 }
35 void insertvalue(int x)
36 {
37     struct trie* temp;
38     temp=root;
39     for(int i=4;i>=0;i--)
40     {
41         bool currentbit=x&(1<<i);
42         if(temp->child[currentbit]==NULL)
43         {
44             temp->child[currentbit]=createnode();
45         }
46         temp=temp->child[currentbit];
47     }
48     temp->value=x;
49 }
50 int xorfinder(int x)
51 {
52     struct trie* temp;
53     temp=root;
54     for(int i=4;i>=0;i--)
55     {
56         bool currentbit=x&(1<<i);
57         if(temp->child[1-currentbit]!=NULL)
58         {
59             temp=temp->child[1-currentbit];
60         }
61         else
62         {
63             temp=temp->child[currentbit];
64         }

```

```
65     }  
66     return (x^temp->value);  
67 }  
68
```