

```

1  /*
2      Author:Frozenscode
3  */
4  #include <bits/stdc++.h>
5  using namespace std;
6  #define pb push_back
7  #define mp make_pair
8  #define fi first
9  #define se second
10 typedef int ll;
11 typedef pair<ll,ll> pii;
12 const ll maxn = 2000010;
13 const ll INF = 2147483647;
14 ll n, a[maxn];
15 struct Trie{
16     ll cnt = 0;
17     ll son[maxn << 2][2];
18     ll l[maxn << 2], r[maxn << 2];
19     void init(){
20         cnt = 0;
21         memset(son, 0, sizeof(son));
22     }
23     void insert(ll x, ll pos){ // root = 0
24         ll rt = 0;
25         for(int i = 30; i >= 0; i--){
26             ll op = (x >> i) & 1;
27             if(!son[rt][op]) son[rt][op] = ++cnt;
28             if(!l[rt]) l[rt] = pos;
29             r[rt] = pos;
30             rt = son[rt][op];
31         }
32         if(!l[rt]) l[rt] = pos;
33         r[rt] = pos;
34     }
35     long long get_ans(ll rt, ll x, ll pos){ // pos 指当前点的高度 0是底层
36         long long ans = 0;
37         for(int i = pos; i >= 0; i--){
38             ll op = (x >> i) & 1;
39             if(son[rt][op]){
40                 rt = son[rt][op];
41             }
42             else{
43                 ans += (1 << i);
44                 rt = son[rt][!op];
45             }
46         }
47         return ans;
48     }
49     void Traceback(int rt){
50         printf("%lld %lld\n", l[rt], r[rt]);

```

```

51     if(son[rt][0]) Traceback(son[rt][0]);
52     if(son[rt][1]) Traceback(son[rt][1]);
53 }
54 long long merge(ll rt, ll pos){
55     if(pos == 0) return 0;
56     if(son[rt][0] && son[rt][1]){
57         long long ans = 2000000000;
58         for(int i = l[son[rt][0]]; i <= r[son[rt][0]]; i++) ans = min(ans, get_ans(son[rt][1], a[i], pos -
2) + (1 << (pos - 1)));
59         return ans + merge(son[rt][0], pos - 1) + merge(son[rt][1], pos - 1);
60     }
61     if(son[rt][0]) return merge(son[rt][0], pos - 1);
62     if(son[rt][1]) return merge(son[rt][1], pos - 1);
63 }
64 }trie;
65 int main()
66 {
67     ios::sync_with_stdio(false);
68     trie.init();
69     cin >> n;
70     for(int i = 1; i <= n; i++) cin >> a[i];
71     sort(a + 1, a + n + 1);
72     for(int i = 1; i <= n; i++) trie.insert(a[i], i);
73     // trie.Traceback(0);
74     cout << trie.merge(0, 31);
75     return 0;
76 }

```