

## <BOJ 10815\_숫자카드>

### 문제

상근이는 숫자 카드  $N$ 개를 가지고 있다. 정수  $M$ 개가 주어졌을 때, 이 수가 적혀있는 숫자 카드를 상근이가 가지고 있는지 아닌지를 구하는 프로그램을 작성

### 입력

$N$  : 상근이가 가진 숫자 카드 개수 ( $1 \leq N \leq 500,000$ )

$N$ 개의 정수 : 상근이의 숫자 카드 (중복 없음)

$M$  : 확인할 숫자 개수 ( $1 \leq M \leq 500,000$ )

$M$ 개의 정수 : 각 숫자가 상근이 카드에 있는지 확인

### 출력

각 숫자에 대해 상근이가 가지고 있으면 1, 없으면 0 출력 (공백 구분)

### 예제 입력 1

```
5
6 3 2 10 -10
8
10 9 -5 2 3 4 5 -10
```

### 예제 출력 1

```
1 0 0 1 1 0 0 1
```

### 접근 방법

상근이가 가진 숫자 카드  $N$ 개를 입력받아  $a$  배열에 저장하고, 이진 탐색을 위해 오름차순으로 정렬한다.  $M$ 개의 숫자를 입력받아  $b$  배열을 순회하며, 각 숫자에 대해 이진 탐색을 수행하여 해당 숫자가 상근이의 카드에 있는지 확인한다. 이진 탐색 결과가 참이면 1, 거짓이면 0을 출력한다.

```

/*
10815/숫자카드/BOJ
208ms
17m
*/
#include<bits/stdc++.h>
using namespace std;

int a[500005];
int b[500005];

int N,M;

int binarySearch(int target){
    int st = 0;
    int en = N-1;
    while(st<=en){
        int mid = (st+en)/2;
        if(a[mid] < target){
            st = mid+1;
        }
        else if(a[mid]>target){
            en = mid-1;
        }
        else return 1;
    }
    return 0;
}

int main(){
    ios::sync_with_stdio(0);
    cin.tie(0);
    cin>>N;
    for(int i=0;i<N;i++){
        cin >> a[i];
    }
    cin>>M;
    for(int i = 0; i<M;i++){
        cin>>b[i];
    }
    sort(a,a+N);

    for(int i=0;i<M;i++){
        cout << binarySearch(b[i]) <<' ';
    }
}

```