#include<bits/stdc++.h>

using namespace std;

long search(vector<long>& nums, long target) {

long left = 0;

long right = nums.size()-1;

while(left<=right){

if(nums[left]==target) return left;

if(nums[right]==target) return right;

long mid = left + (right-left)/2;

if(nums[mid]==target) return mid;

if(nums[left]>nums[mid]){

// rotation has taken place before or at mid

// 4 5 0 1 2 mid = 0 tar=1

if(target>nums[mid] && target<nums[left]){

left = mid+1;

}

else{

right = mid-1;

}

}

else{

// rotation has not taken place till now

// 4 5 6 7 0 1 2 mid = 7 tar=6

if(target>nums[left] && target<nums[mid]){

right = mid-1;

}

else{

left = mid + 1;

}

}

}

return -1;

}

int main(){

long n,q;

cin>>n>>q;

vector<long> arr(n);

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

cin>>arr[i];

}

while(q--){

long num;

cin>>num;

long ans = search(arr,num);

if(ans!=-1){

cout<<ans+1<<endl;

}

else{

cout<<"-1"<<endl;

}

}

}