Alicia y las llaves doradas de las puertas

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos | 11.08 | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |
| Límite de entrada (bytes) | 10 KiB |  |  |

**Descripción**

Después de su larga caída, Alicia quedó atrapada en el fondo del agujero del conejo. Afortunadamente no estaba totalmente oscuro y pudo ver que había muchas puertas (todas cerradas) y una mesa con varias llaves doradas. Seguramente alguna llave abriría alguna de estas puertas, así que Alicia lo intentó y pronto descubrió que las chapas de las puertas eran de diferentes tamaños y que estaban ordenadas en fila, de la más pequeña a la más grande. Cada llave sólo abriría la chapa del mismo tamaño. Como en verdad eran muchas puertas y muchas llaves, Alicia pensó que le tomaría mucho tiempo intentar abrir todas las puertas con las llaves correctas.

**Entrada**

Un entero MM seguido de los tamaños de las chapas P1,...,PMP1,...,PM. Posteriormente un entero NN seguido de los tamaños de las llaves L1,...,LNL1,...,LN. Puedes suponer que 1≤N,M≤100,0001≤N,M≤100,000, que los tamaños de las chapas son distintos con 1≤P1<P2<...<PM≤100,0001≤P1<P2<...<PM≤100,000 y que los tamaños de las llaves cumplen 1≤Li≤100,0001≤Li≤100,000 para toda 1≤i≤N1≤i≤N.

**Salida**

Para cada una de las NN llaves, el número de la puerta que puede ser abierta con esa llave o 00 si no corresponde con ninguna puerta.

**Ejemplo**

| **Entrada** | **Salida** |
| --- | --- |
| 5  1 3 4 5 9  4  2 5 1 8 | 0 4 1 0 |

*Fuente: UAM Azcapotzalco 2016*

Problema subido por: [rcc](https://omegaup.com/profile/rcc/)

[Reportar contenido inapropiado en este problema.](https://omegaup.com/arena/problem/aldp)

| **Enviado** | **GUID** | **Estatus** | **Porcentaje** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/aldp#problems/new-run) | | | | | | | | |
| 2019-01-16 11:44:12 | 62ff81a4 | Respuesta correcta | 100.00% | cpp | 9.35 MB | 0.46 s |  |  |
| 2019-01-16 05:45:13 | 4945f11f | Tiempo límite excedido | 90.00% | cs | 22.85 MB | >2.56 s |  |  |
| 2019-01-16 05:32:00 | b5168e4d | Error de compilación | — | cs | — | — |  |  |
| **Envíos** | | | | | | | |

<https://omegaup.com/arena/problem/aldp#problems>

#include <iostream>

#include <stdio.h>

#include <set>

#include <map>

#include <vector>

using namespace std;

int main() {

int m;

scanf("%d",&m);

set<int> p;

std::vector<int> vector\_p;

for(int i =0; i<m; i++) {

int elem;

scanf("%d", &elem);

p.insert(elem);

vector\_p.push\_back(elem);

}

int n;

scanf("%d", &n);

int l[n];

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

int elem;

scanf("%d", &elem);

l[i] = elem;

}

std::map<int,int> indices;

for(int i =0; i<m; i++) {

indices[vector\_p[i]] = i + 1;

}

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

if(indices.find(l[i]) != indices.end()) {

cout << indices[l[i]] << " ";

}

else{

cout << "0 ";

}

}

return 0;

}