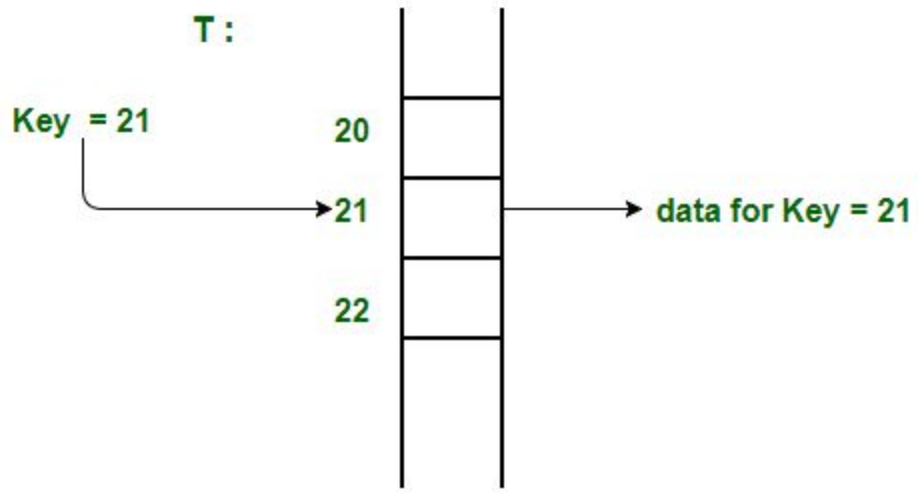
Index Mapping (Trivial Hashing) with negative values allowed

Task: Given a limited range array that contains both positive and non-positive numbers, i.e., elements that are in the [-MAX,MAX] range, our task is to serch if some number is present in the array or not in O(1) time.

Reference:
https://www.geeksforgeeks.org/index-mapping-or-trivial-hashing-with-negatives-allowed/

Given the fact that the range is limited, we can use index mapping (or trivial hashing). We will use the values as the index in a big array and thus if we do so, O(1) time is ensured.

The following image explains the basic concept:



However, how will we deal with negative numbers? A solution is to use a 2D array of size hash[MAX+1][2]

Pseudocode outline:

```
Assign all the values of the hash matrix as 0.
Traverse the given array:
    If the element ele is non negative assign
        hash[ele][0] as 1.
    Else take the absolute value of ele and
         assign hash[ele][1] as 1.
```

To search any element x in the array.

- If X is non-negative, then check if hash[X][0] is 1 or not. If hash[X][0] is 1 then the number is present (or it is not present otherwise).
- If X is negative, then take the absolute value of X and then check if hash[X][1] is 1 or not. If hash[X][1] is 1 then the number is present.

Provide an implementation of this simple way of hashing.