```
class Solution {
    public int[] intersect(int[] nums1, int[] nums2) {
        Map<Integer, Integer> mp = new HashMap<>();
        for (int i : nums1) {
            int v = mp.getOrDefault(i, 0);
            mp.put(i, v+1);
        }
        ArrayList<Integer> list = new ArrayList<>();
        for (int i : nums2) {
            if (mp.containsKey(i)) {
                list.add(i);
                if (mp.get(i) == 1) {
                    mp.remove(i);
                } else {
                    mp.put(i, mp.get(i)-1);
                }
            }
        }
        int[] ret = new int[list.size()];
        for (int i = 0; i < list.size(); i++){
            ret[i] = list.get(i);
        }
        return ret;
    }
}
```

- What if the given array is already sorted? How would you optimize your algorithm?
- What if *nums1*'s size is small compared to *nums2*'s size? Which algorithm is better?
- What if elements of *nums2* are stored on disk, and the memory is limited such that you cannot load all elements into the memory at once?