Cursors

- → Enumeration
- → Iterator ✓
- → ListIterator

Iterable → Parent Interface of Collection

① Enumeration

Steps ① Create Enumeration Cursor

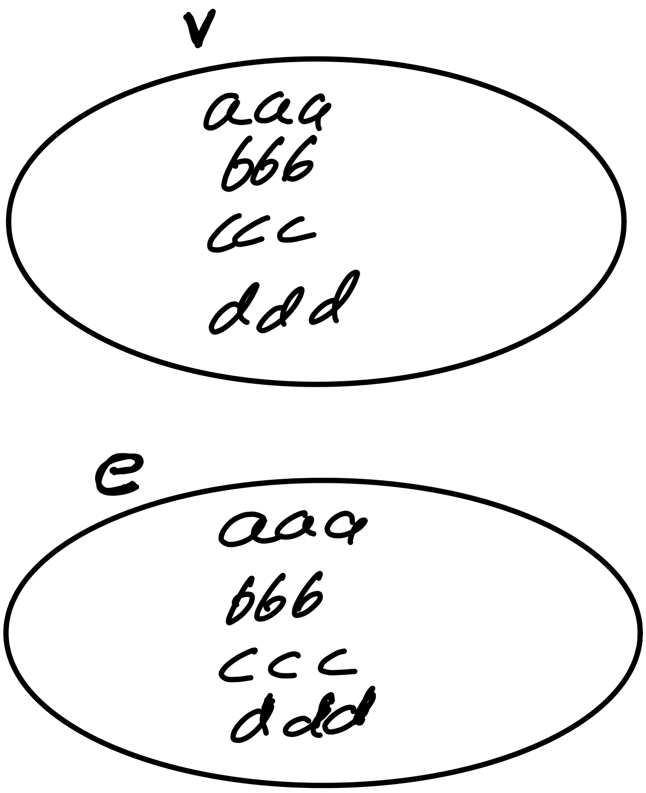
↳ elements() ⇒ Vector/Stack

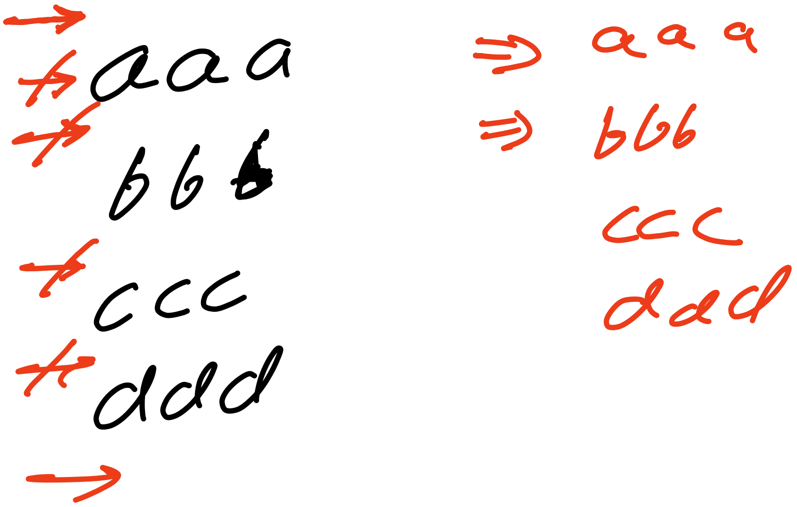
⇓

return type ⇒ Enumeration

② hasMoreElements()
nextElement() } → In Enumeration

```
public class Test1 {
    public static void main(String[] args) {
        Vector v = new Vector() ;
        v.add("aaa") ;
        v.add("bbb") ;
        v.add("ccc") ;
        v.add("ddd");

        Enumeration e = v.elements() ;
        while(e.hasMoreElements()){
            System.out.println(e.nextElement());
        }
    }

}
```





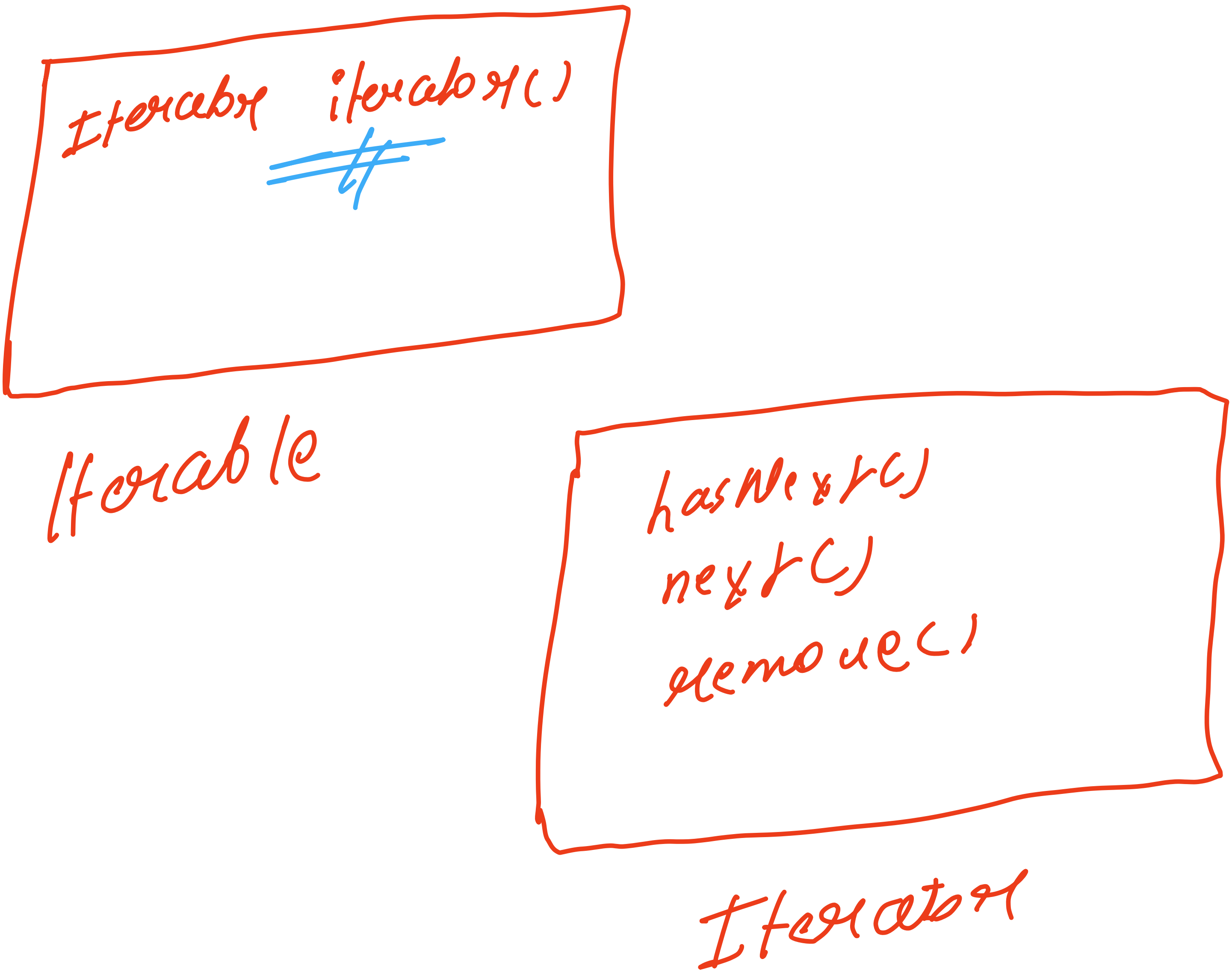
read ✓

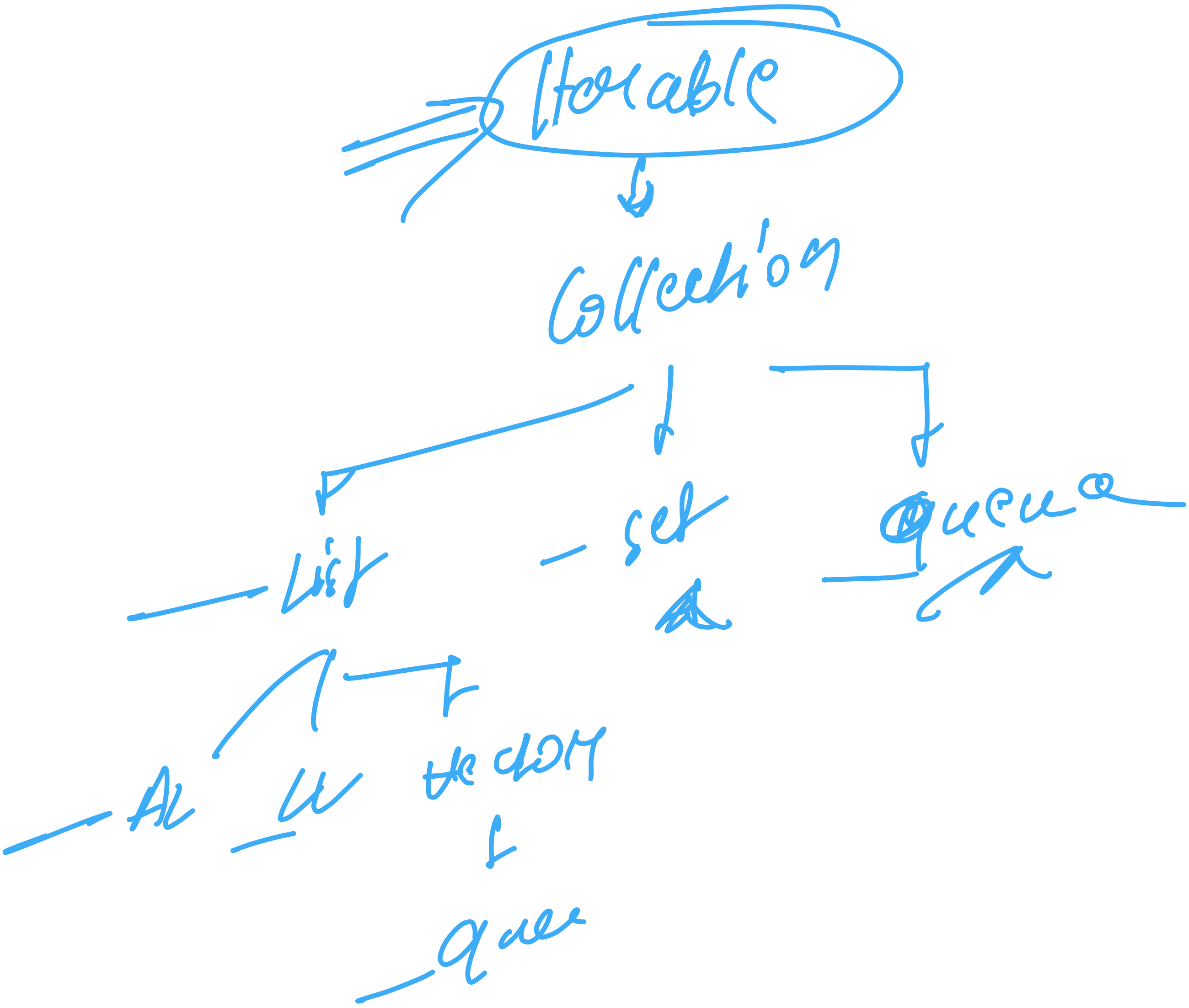
update/remove X

Iterator

① Creation Iterator Cursor

Iterator iterator()





② read one by one

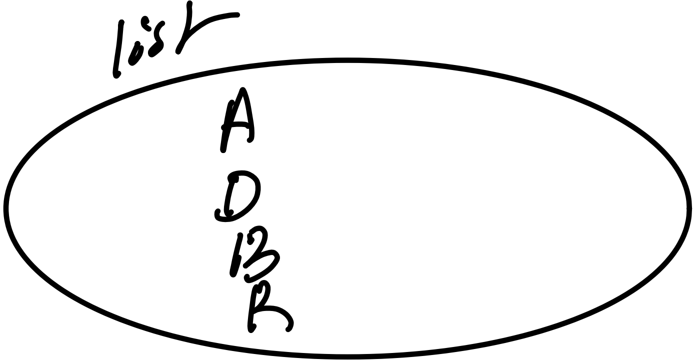
ListIterator

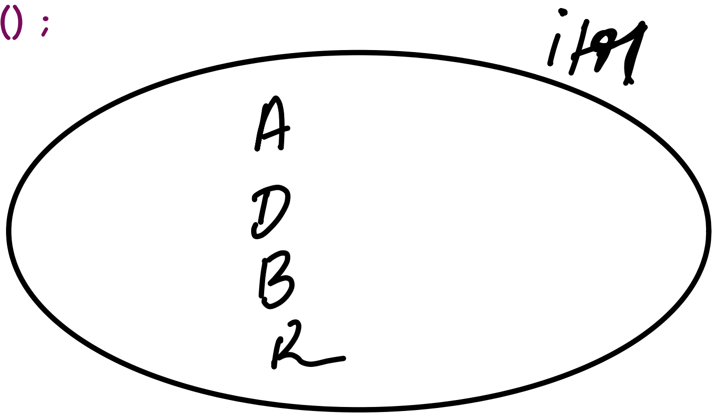
① ListIterator listIterator()

② read one by one

- hasNext()
- next()
- set(—)
- add(→)
- hasPrevious()
- previous()
- nextIndex();
- remove();

```
ArrayList<String> list = new ArrayList<>() ;
        list.add("Arun") ;
        list.add("Divas") ;
        list.add("Bhavek") ;
        list.add("Rahul") ;

        ListIterator<String> itr = list.listIterator() ;

2       while(itr.hasNext()){
            System.out.println(itr.next());
        }

        System.out.println("===============");

1       while(itr.hasPrevious()){
            System.out.println(itr.previous());
        }
```





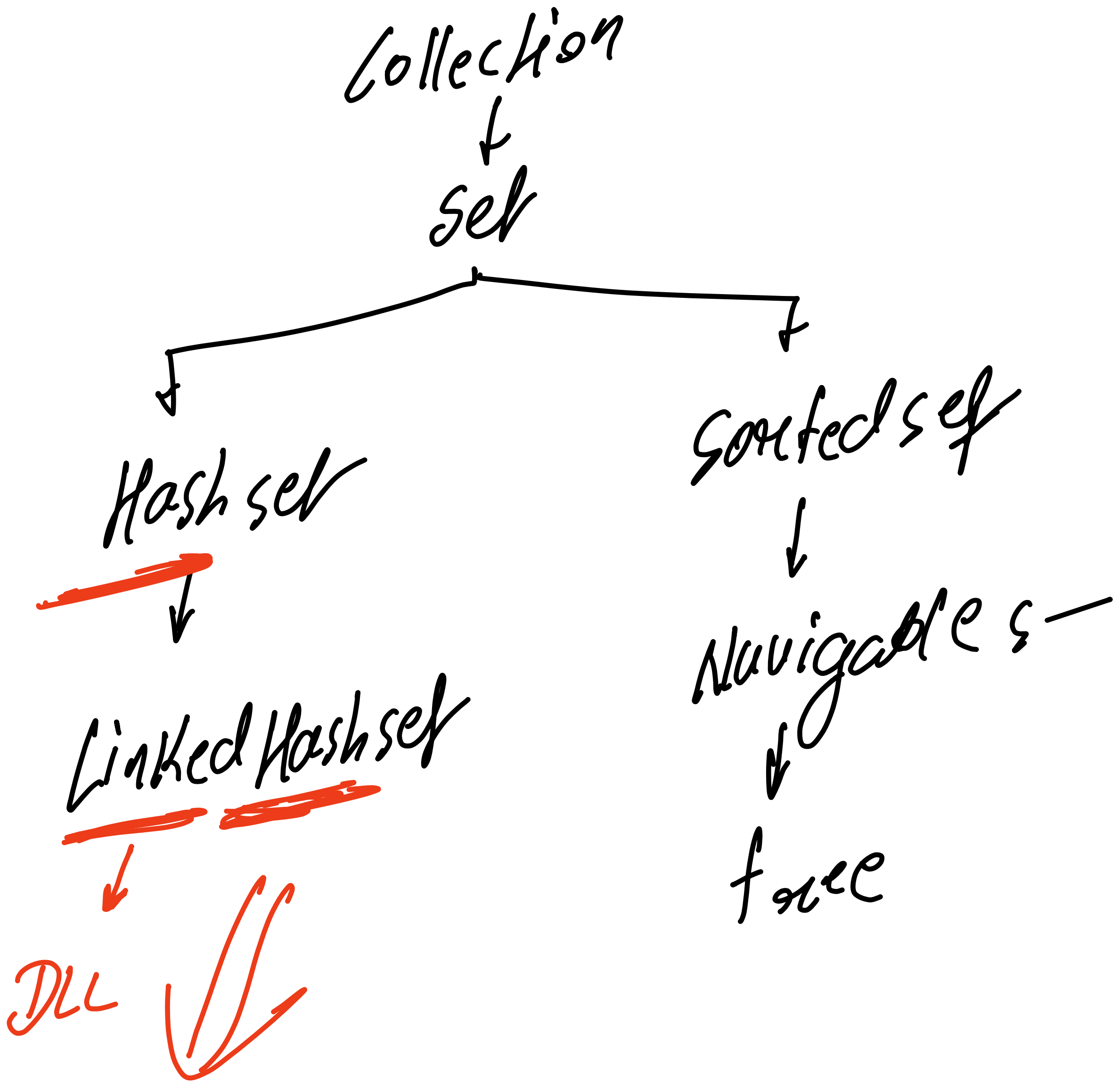
→

A
D
B
R

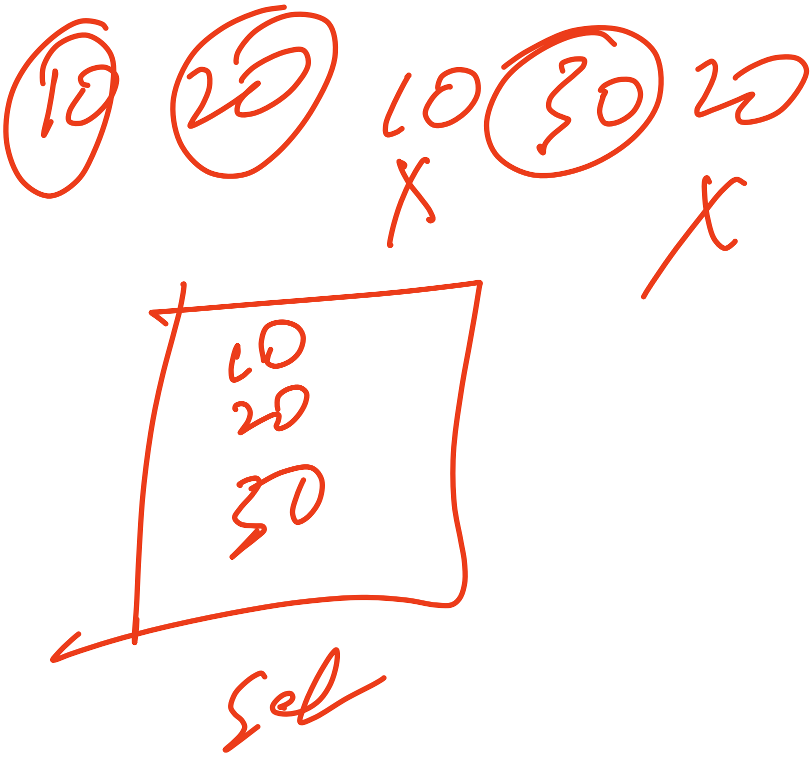
A
D
B
R
- - - -
R
B
D
A

Set

- → Interface
- → 1.2 V



DLL + Hashset



Hashset

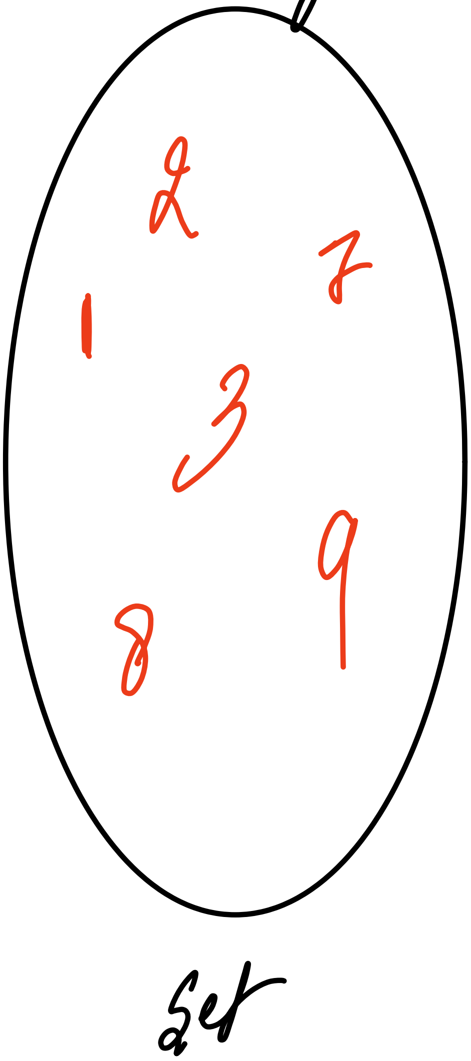
$$\text{load factor} \Rightarrow \frac{\text{no of nodes}}{\text{size of Array}}$$

↓

0.75

Q: $A = \{1, 2, 1, 3, 7, 8, 2, 9\};$

find no. of distict element.



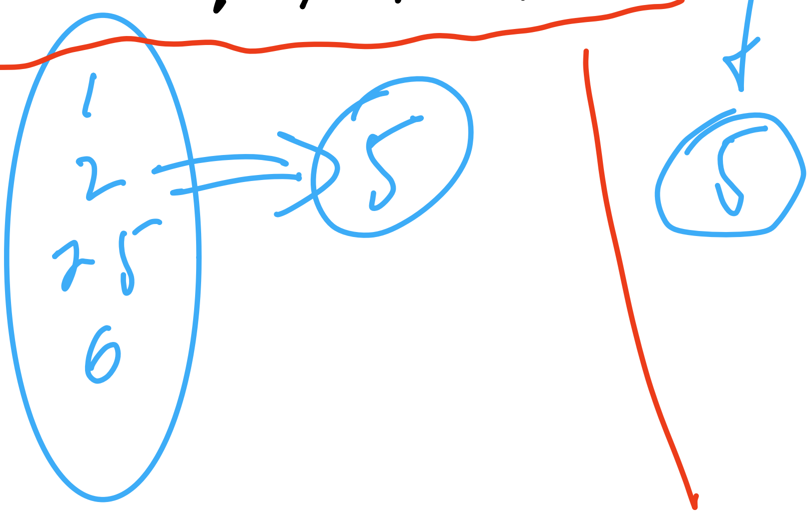
$\Rightarrow \boxed{6}$

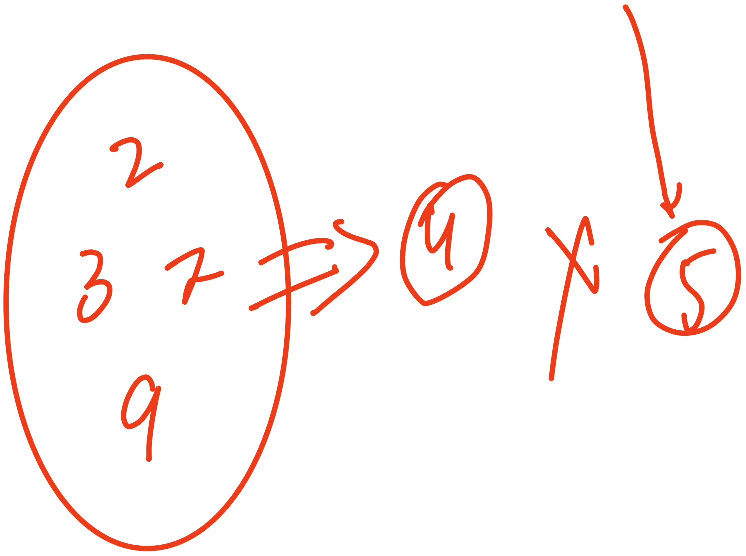
```
HashSet<Integer> HS = new HashSet<>();
for(i=0 ; i<N; i++){
    int ele = A[i];
    set.add(ele);
}
```

check if all the elements are Distinct or not

A = 1, 2, 7, 5, 6 → True

A = 2, 3, 7, 9, 3 → false




2
3
7
9
⇒
4
×
5