أسئلة على السيت والماب

شعبة 3

محمد ابو صفط

Q1: given an array, determine if it contains duplicates or not.

```java
public static boolean hasDuplicates(int[] array) {
    HashSet<Integer> set = new HashSet<>();
    for (int i : array) set.add(i);
    return array.length != set.size();
}
```

Q2: inside MyLinkedList class implement a method of type boolean, return true if the list has a cycle otherwise return false.

answer inside MyLinkedList class

```java
public boolean hasCycel() {
    HashSet<Node<E>> set = new HashSet<>();
    Node<E> temp = head;
    while (temp != null) {
        if (set.contains(temp))
            return true;
        set.add(temp);
        temp = temp.next;
    }
    return false;
}
```

Q3: given a String s, print the first unique character in it, or print -1 if there is no unique characters.

```java
public static void PrintFirst(String s){
    Map<Character,Integer> map = new LinkedHashMap<>();
    for (int i = 0; i < s.length(); i++) {
        if(!map.containsKey(s.charAt(i)))
            map.put(s.charAt(i), 1);
        else map.replace(s.charAt(i) , map.get(s.charAt(i))+1);
    }
    for( Character x : map.keySet()){
        if(map.get(x)==1) {
            System.out.println(x);
            return;
        }
    }
    System.out.println(-1);
}
```

. Q4: given two Array Lists, return an Array List that contains the the intersection of the two lists. Note: the intersection may contain duplicates.

```java
public static ArrayList<Integer> Dub(ArrayList<Integer> A1, ArrayList<Integer> A2) {
    Map<Integer, Integer> m1 = new HashMap<>();
    Map<Integer, Integer> m2 = new HashMap<>();
    for (Integer x : A1) {
        if (!m1.containsKey(x))
            m1.put(x, 1);
        else m1.put(x, m1.get(x) + 1);
    }
    for (Integer x : A2) {
        if (!m2.containsKey(x))
            m2.put(x, 1);
        else m2.put(x, m2.get(x) + 1);
    }
    ArrayList<Integer> A = new ArrayList<>();

    for (Integer z : m1.keySet()) {
        if (m2.containsKey(z)) {
            for (int i = 0; i < Math.min(m1.get(z), m2.get(z)); i++) {
                A.add(z);
            }
        }
    }
    return A;
}
```

ميثود بستقل اريه من انتجر وبكون العناصر بيها غير متكررة وبستقبل كمان تارجت*
انتجر برضو رجعلي ترو اذا كان موجود عنصرين حاصل جمعهم بساوي التار جت
ب حل التايم كومبلكسيتي اله
O(N

```java
public static boolean twoSum(int[] A, int target) {
    Set <Integer> set =new HashSet<>();
    for (int i = 0; i < A.length; i++)
        set.add(A[i]);

    for(int i = 0; i < A.length; ++i){
        set.remove(A[i]);
        if(set.contains(target - A[i])){
            return true ;
        }
        set.add(A[i]);
    }
    return false;
}
```