BİR LİSTE İÇİNDE TEKRAR EDEN ÖĞELERİ BULMA

BİR LİSTE İÇİNDE TEKRAR EDEN ÖĞELERİ BULMA

Bir liste içinde tekrar eden öğeleri bulma(Finding Duplicate Elements in a List) algoritması problemin adından da belli olduğu üzere listedeki tekrar eden elemanların çeşitli nedenlerden dolayı bulma işlemini yapar. Bu nedenler bir göz atalım.

Veri Analizi ve Temizliği: Veri biliminde ve analizinde, tekrar eden öğeleri belirlemek verinin doğruluğunu ve tutarlılığını sağlamak için önemlidir.

Performans Optimizasyonu: Tekrar eden öğeleri bulmak ve bunları etkin bir şekilde ele almak, veri işleme sürelerini kısaltabilir ve algoritma performansını artırabilir.

Kaynak Kullanımının Azaltılması: Büyük veri kümeleriyle çalışırken, tekrar eden öğelerin bulunması ve bunların dikkate alınması, bellek ve işlemci gücü kullanımını azaltabilir.

Veri Görselleştirme ve Anlaşılabilirlik: Tekrar eden öğelerin belirlenmesi, veri setinin yapısal özelliklerini daha iyi anlamamıza ve görselleştirmemize yardımcı olabilir.

Bir liste içinde tekrar eden öğeleri bulmak gerçek hayatta nerelerde karşımıza çıkar?

Bu problemin karşımıza çıkabileceği sonsuz senaryolar mevcuttur.
Biz sizin için üç tanesini belirledik ve çözümü için gerekli sudo kod ve java kodlarını yazdık.

1)Bir grupta doğum yılı aynı olan kişiler: Yaş hesaplaması yapılırken kullanılabilir
(int değişken)

2) Bir cümlede tekrar eden harfleri bulan program(char değişken)

3)Bir dükkandaki malzeme sayımı(String değişken)

```
Function findDuplicates(list)

evenDuplicates=neW empty array

startIndex=0

for i=0 to list.length do

for j=i+1 to list.length do

if list[i]==list[j]

if not isFind(evenDuplicates,list[i])

evenDuplicates[startIndex++]=list[i]


end  if

end if

end for

end for

return evenDuplicates

end function
```

```
Function isFind(arr,value)

For i=0 to arr.length

if arr[i] ==value

Return true

End if

End for

Return false

End function
```

Listedeki aynı doğum yıllarını bulan kod

```java
import java.util.*;

public class BirGruptaDogumYiliAyniOlanKisiler {
    public static boolean isFind(int[] list, int value) {  1 usage
        for (int i = 0; i < list.length; i++)
            if (list[i] == value) {
                return true;
            }
        return false;
    }

    /* isFind metodunu eğer bir tane bulmuşsak tekrar tekrar döndürüp startIndex değerini arttırmamak için oluşturduk. boolean tipindeki
     metodumuz indexteki sayı girilen değere eşitse true, değilse false döndürüyor */

    public static int[] findDuplicates(int[] list) {  1 usage
        int[] duplicates = new int[list.length]; //tekrar eden sayıları atamak için, duplicates adında bir array oluşturduk
        int startIndex = 0; //array içine atama yapmak için startIndex adında bir değişken oluşturduk

        for (int i = 0; i < list.length; i++) {
            for (int j = i + 1; j < list.length; j++) {
                if (list[i] == list[j]) {
                    if (!isFind(duplicates, list[i])) {
                        duplicates[startIndex] = list[i];
                        startIndex++;
                    }
                }
            }
        }

        /* iki for döngüsüyle list adındaki array içerisinde gezdik, birden fazla olan elemanları oluşturduğumuz duplicates adındaki array'e kopyaladık.
         isFind metoduyla da var mı yok mu kontrol ederek tekrar tekrar kopyalamasını engelledik */

        return Arrays.copyOf(duplicates, startIndex);
```

```java
        /* arrays classından copyOf metodunu çağırarak duplicates arrayinin startIndex değişkeni kadar yani tekrar edenlerin size'ı kadar sonucu döndürdük. eğer
        sadece duplicates olarak çağırsaydık yukarıda list.length olarak size tanımladığımız için geri kalanları 0 olarak döndürecektik. biz
        copyOf metoduyla bunun önüne geçtik */
    }

    public static void main(String[] args) {
        Random random = new Random(); //random classını tanımladık
        int[] list = new int[10]; //grubumuzu 20 kişi olarak belirledik

        for (int i = 0; i < list.length; i++) { //for döngüsüyle indisi tek tek geziyoruz
            list[i] = random.nextInt( origin: 1,  bound: 10); //elemanların rastgele atanması için random classını kullandık, aralık belirledik
        }


        System.out.println(Arrays.toString(list)); //elemanları yazdırmak için arrays classından toString metodunu çağırdık

        int[] array = findDuplicates(list); //metodun çıktısını yazdırabilmek için bir diziye atadık

        System.out.println("grupta birden fazla olan doğum yılları: " + Arrays.toString(array)); //sonucunu yazdırdık

    }
}
```

```
[1993, 1993, 1993, 1993, 1993, 1995, 1996, 1997, 1997, 1997, 1998, 1998, 2000, 2000, 2001, 2002, 2002, 2002, 2002, 2003]
grupta birden fazla olan doğum yılları: [1993, 1997, 1998, 2000, 2002]

Process finished with exit code 0
```

Bir cümlede tekrar eden harfleri bulan kod

```
lütfen bir cümle girin
Başka bir evrende en güzel halinle..
tekrar eden karakterler:[a, i, r, e, n, l, .]
```

```java
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
public class Tekraredenharf {
    public static  void main(String[] args){
        Scanner scan=new Scanner(System.in);
        System.out.println("lütfen bir cümle girin");
        String cumle =scan.nextLine(); //cümle kullanıcıdan aldık.
        tekraredenkarakterleribulanmetod(cumle);

    }
    1 usage
    public static void tekraredenkarakterleribulanmetod(String str){// tek parametreli bir metot olusturduk.
        List<String> list =new ArrayList<>(); //tekrar eden harfleri topladığımız bir liste olusturduk.
        int sayac=0;
        for(int i=0;i<str.length();i++) {// birinci indeksten başlayıp eşit bulunca sayacı bir artıracak.
            for (int j = i + 1; j < str.length(); j++) {// str.length() cumlenın sonuna kadar tekrar et
                if (str.charAt(i) == str.charAt(j)) {
                    sayac++;
                }
            }
        if(sayac>0 && !list.contains(str.substring(i,i+1)) && !str.substring(i,i+1).contains(" ")){ // tekrar eden harfleri koyduğumuz listeye
            list.add(str.substring(i,i+1));// tekrar eden harfleri burda liste yerleştirdik.
        }
        sayac=0; // yukarı döngüye dönmeden önce sayacı başlangıç değerini sıfıra eşitledik
    }
    System.out.println("tekrar eden karakterler:"+ list);
}}
//kullanıcının girmiş olduğu bir stringde tekrarlanan karakterleri  bulan ve consola yazdıran metod kod.
```

Malzeme sayımında birden fazla olan malzemeleri veren kod

```java
import java.util.List;

public class malzemeSayimi {
    public static void main(String[] args) {

List<String> products = new ArrayList<>(); // Ürün listesi oluşturuluyor ve ürünler ekleniyor.
        products.add("Un");
        products.add("Ekmek");
        products.add("Ekmek");
        products.add("Yag");
        products.add("Sut");
        products.add("Makarna");
        products.add("Yag");
        products.add("Sut");
        products.add("Ekmek");
        products.add("Makarna");
        products.add("Ekmek");
        products.add("Sut");
        products.add("Sut");
        products.add("Makarna");
```

```java
    while (products.size() > 0) {  // Listede ürün bitene kadar döngünün devam etmesini sağlıyor.
        String product = products.get(0); // Listeden ilk ürün alınıyor.
        int count = 1; // Her üründen en az 1 adet olduğu için sayaç 1'den başlatılıyor.
        products.remove( index: 0); // Bu ürünün sayımı yapıldığı için listeden çıkarılıyor.

        for (int i = 0; i < products.size(); i++) { // Bu ürünle aynı olan ürünleri bulmak için bütün liste taranıyor.

            if (product.equals(products.get(i))) { // i'nci ürün bu ürünle aynıysa
                count++; // Sayaç 1 arttırılıp
                products.remove(i); // Eşleşen ürün listeden çıkarılıyor.
                i--; // Listeden bir ürün kaldırıldığı için indeks geri alınır.
            }
        }
        if (count > 1) {
            System.out.println(count + " adet " + product + " bulunmaktadir.");
        }
    }
}
```

```
4 adet Ekmek bulunmaktadir.
2 adet Yag bulunmaktadir.
4 adet Sut bulunmaktadir.
3 adet Makarna bulunmaktadir.
```

Teşekkürler

Hazırlayanlar

ESRA AKTULUM

ŞİFANUR ÖNAL

SÜMEYYE ELGİN

ERVA BAYBURTLU