

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

```
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 bulmussak 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);
```



```
/* arrays classından copyOf metodunu çağırarak duplicates arrayinin startIndex değışkeni 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[20]; //grubumuzu 20 kişi olarak belirledik  
  
    for (int i = 0; i < list.length; i++) { //for döngüsüyle indisi tek tek geziyoruz  
        list[i] = random.nextInt( origin: 1992, bound: 2004); //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

```
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("l\u00fctfen bir c\u00fcmle girin");
        String cumle =scan.nextLine(); //c\u00fcmle kullan\u0131c\u0131dan ald\u0131k.
        tekraredenkarakterleribulanmetod(cumle);
    }
```

l\u00fctfen bir c\u00fcmle girin

Ba\u015fka bir evrende en g\u00fczel halinle..

tekrar eden karakterler:[a, i, r, e, n, l, .]

1 usage

```
public static void tekraredenkarakterleribulanmetod(String str){// tek parametrel\u0131 bir metot olu\u015fturduk.
    List<String> list =new ArrayList<>(); //tekrar eden harfleri toplad\u0131\u0131m\u0131z bir liste olu\u015fturduk.
    int sayac=0;
    for(int i=0;i<str.length();i++) {// birinci indeksten ba\u015flay\u0131p e\u015fit bulunca sayacı bir art\u0131racak.
        for (int j = i + 1; j < str.length(); j++) {// str.length() cumlenin 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 koydu\u011fumuz listeye
            list.add(str.substring(i,i+1));// tekrar eden harfleri burda liste yerle\u015ftirdik.
        }
        sayac=0; // yukarı d\u00f6ng\u00feye d\u00f6nmeden \u00f6nce sayacı ba\u015lang\u0131\u00e7 de\u011ferini s\u0131fıra e\u015itledik
    }
    System.out.println("tekrar eden karakterler:"+ list);
}
}
```

//kullan\u0131c\u0131n\u0131n girmi\u015f oldu\u011fu bir stringde tekrarlanan karakterleri bulan ve consola yazdıran metod kod.

Malzeme sayımında birden fazla olan malzemeleri veren kod

```
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");
```



```

while (products.size() > 0) { // Listede ürün bitene kadar döngünün devam etmesini sağlıyor.
    String product = products.get(0); // Listedeki ilk ürün alınıyor.
    int count = 1; // Her üründen en az 1 adet olduğu için sayacı 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++; // Sayacı 1 arttırılıp
            products.remove(i); // Eşleşen ürün listeden çıkarılıyor.
            i--; // Listedeki bir ürün kaldırıldığı için indeks geri alınır.
        }
    }
    if (count > 1) {
        System.out.println(count + " adet " + product + " bulunmaktadır.");
    }
}
}

```

4 adet Ekmek bulunmaktadır.
 2 adet Yağ bulunmaktadır.
 4 adet Süt bulunmaktadır.
 3 adet Makarna bulunmaktadır.

Teşekkürler

Hazırlayanlar

ESRA AKTULUM

ŞİFANUR ÖNAL

SÜMEYYE ELGİN

ERVA BAYBURTLU

KAYNAKÇA

<https://www.baeldung.com/java-list-find-duplicates>

<https://chat.openai.com/>

<https://www.geeksforgeeks.org/find-duplicates-given-array-elements-not-limited-range/>