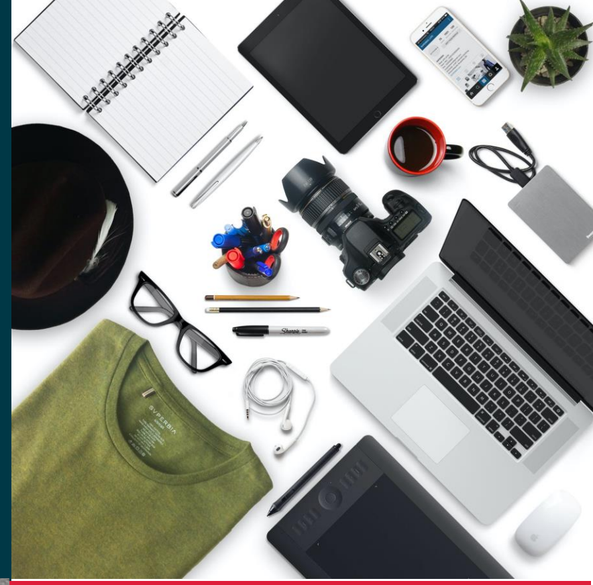
PDP ONLINE

DSA – Data Structures

ArrayList



Deficiency of Array

```java
public class Main {

    public static void main(String[] args) {

        int[] numbers = new int[3];
        numbers[0] = 5;
        numbers[1] = 1;
        numbers[2] = 3;
        numbers[3] = 7;

    }
```

oblems @ Javadoc Declaration Console

inated> Main (1) [Java Application] /Users/user/.p2/pool/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.macosx.

```
eption in thread "main" java.lang.ArrayIndexOutOfBoundsException:
    at Main.main(Main.java:33)
```

```java
public class Main {

    public static void main(String[] args) {

        int[] numbers = new int[3];
        numbers[0] = 5;
        numbers[1] = 1;
        numbers[2] = 3;
        //numbers[3] = 7;

        System.out.println(numbers);
        System.out.println(Arrays.toString(numbers));
    }
```

oblems @ Javadoc Declaration Console

inated> Main (1) [Java Application] /Users/user/.p2/pool/plugins/org.eclipse.justj.openjdk.hotspo

```
[I@dcf3e99
[5, 1, 3]
```

java.util.ArrayList

```java
public static void main(String[] args) {

    ArrayList list = new ArrayList();
    list.add(12);
    list.add(2);
    list.add(4);
    list.add(78);

    list.remove(2);
    list.indexOf(4);

    System.out.print(list);
}
```

Create ArrayList

```
public class ArrayList {

    private int[] items;
    private int count;


    public ArrayList(int length){
        items = new int[length];
    }

    public void print() {
        for(int i=0; i<count; i++) {
            System.out.println(items[i]);
        }
    }
```

insert

```
public void insert(int item) {
    //resize items
    if(items.length == count) {
        int[] newItems = new int[count * 2];
        for(int i=0;i<count; i++) {
            newItems[i] = items[i];
        }
        items = newItems;
    }
    //add item
    items[count++] = item;
}
```

indexOf

```
public int indexOf(int item) {
    for(int i=0;i<count; i++) {
        if(items[i] == item) {
            return i;
        }
    }
    return -1;
}
```

insertAt

```
public void insertAt(int item, int index) {

    //resize items
    if(items.length == count) {
        int[] newItems = new int[count * 2];
        for(int i=0;i<count; i++) {
            newItems[i] = items[i];
        }
        items = newItems;
    }
    // shift items
    for(int i=count-1; i>=index; i--) {
        items[i+1] = items[i];
    }
    //add item
    count++;
    items[index] = item;
}
```

iremoveAt

```
public void removeAt(int index) {
    //invalid index
    if(index < 0 || index >= count) {
        throw new IllegalArgumentException();
    }
    // shift items
    for(int i=index; i<count; i++) {
        items[i] = items[i+1];
    }
    count--;
}
```

Deficiency of ArrayList

```java
public void removeAt(int index) {
    //invalid index
    if(index < 0 || index >= count) {
        throw new IllegalArgumentException();
    }
    // shift items
    for(int i=index; i<count; i++) {
        items[i] = items[i+1];
    }
    count--;
}
```

Task 1
Darsda o`tilgan ArrayList ning Insert, InsertAt, IndexOf, RemoveAt funksiyalari uchun Time Complexity larini aniqlang.

Task 2
Darsda o`tilgan ArrayList ning Maximum qiymatini topadigan funksiya yarating hamda uning Time Complexity sini aniqlang.

public void max()

Task 3
Darsda o`tilgan ArrayList ning Minimum qiymatini topadigan funksiya yarating hamda uning Time Complexity sini aniqlang.

public void min()

Task 4
Darsda o`tilgan ArrayList ning qiymatlarini teskariga almashtiradigan Reverse funksiya yarating hamda uning Time Complexity sini aniqlang.

public void reverse()

Task 5
Darsda o`tilgan ArrayList ning qiymatlari va berilgan array ning qiymatlari orasida umumiy bo`lgan qiymatlarni array shaklida qaytaradigan Intersect funksiya yarating hamda uning Time Complexity sini aniqlang.

public int[] intersect(int[] arr)