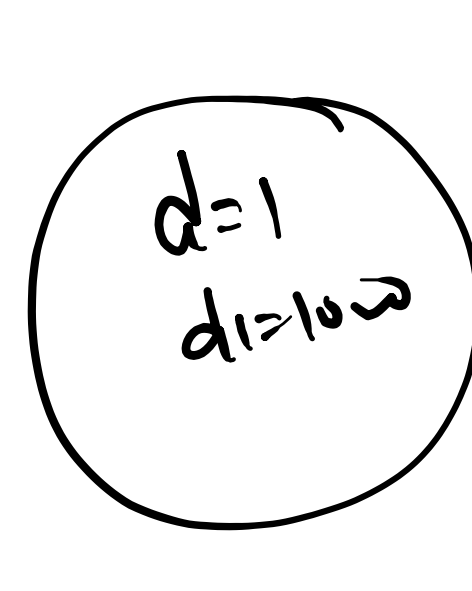
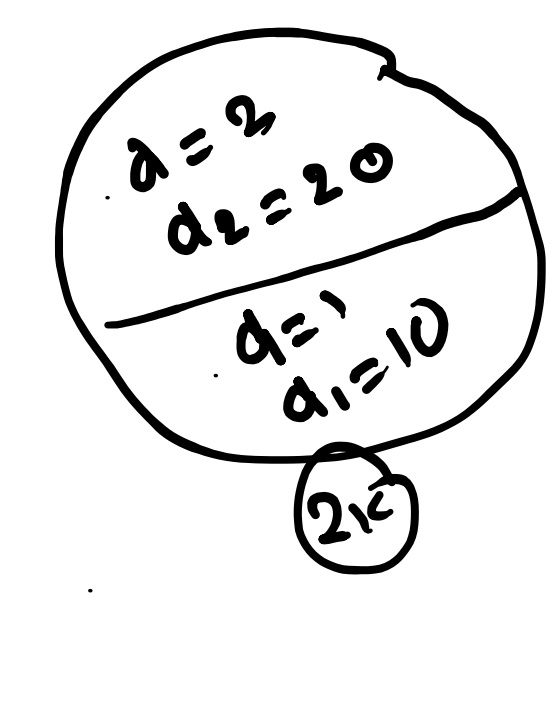
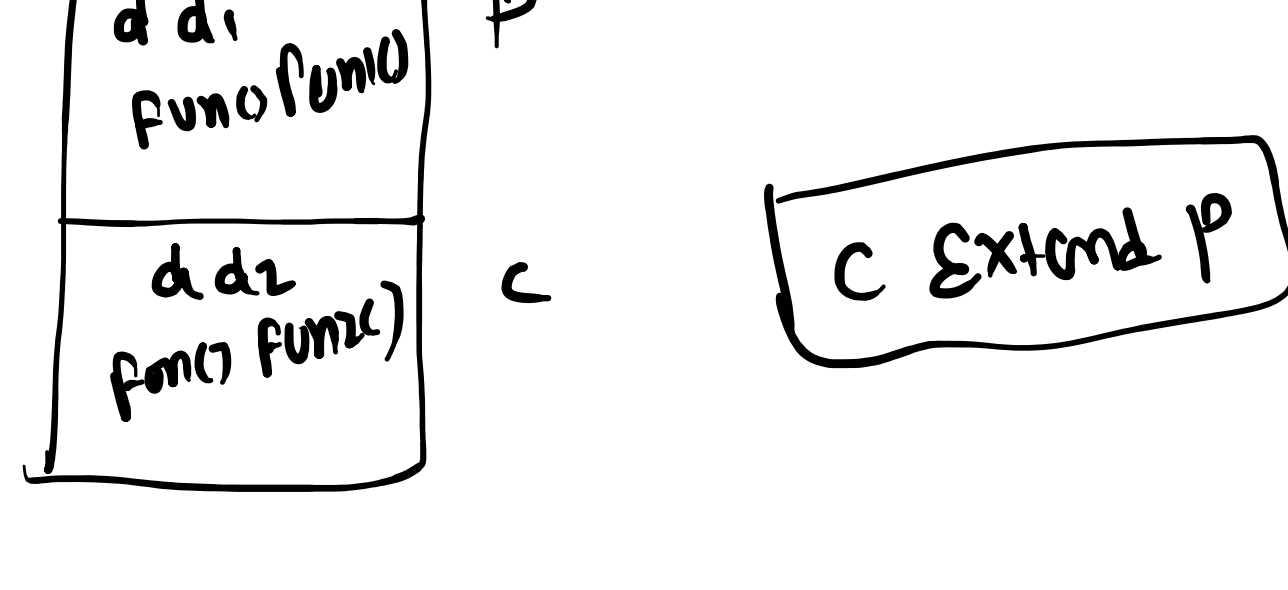


L.H.S	R.H.S
ClassName variable	= new ClassName()
① P obj	new P() -> x
② P obj	new C() -> new ArrayList C()
Ex <- List<> ll	new P() ✓
③ C obj	new C()
④ C obj	

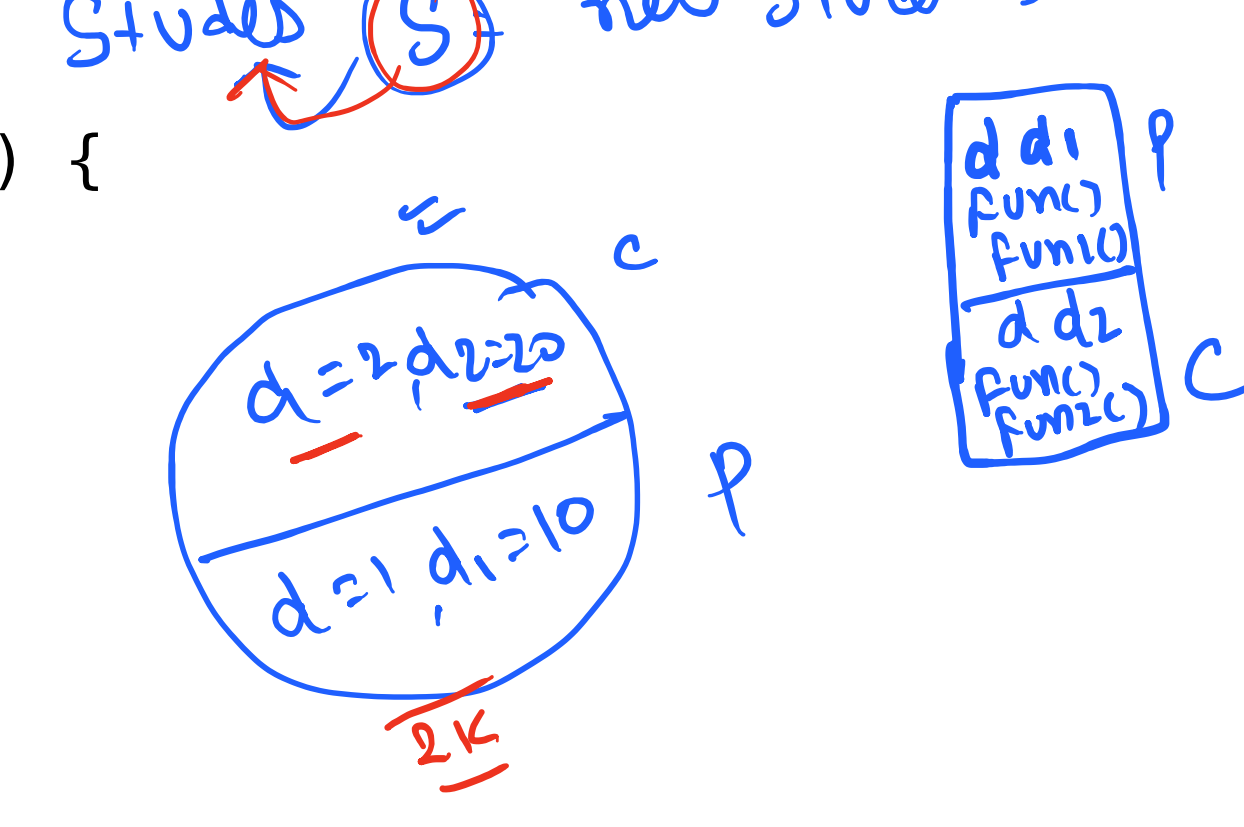
```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    // Case - 1
    P obj = new P();
    System.out.println(obj.d);
    System.out.println(obj.d1);
    obj.fun();
    obj.fun1();
}
```



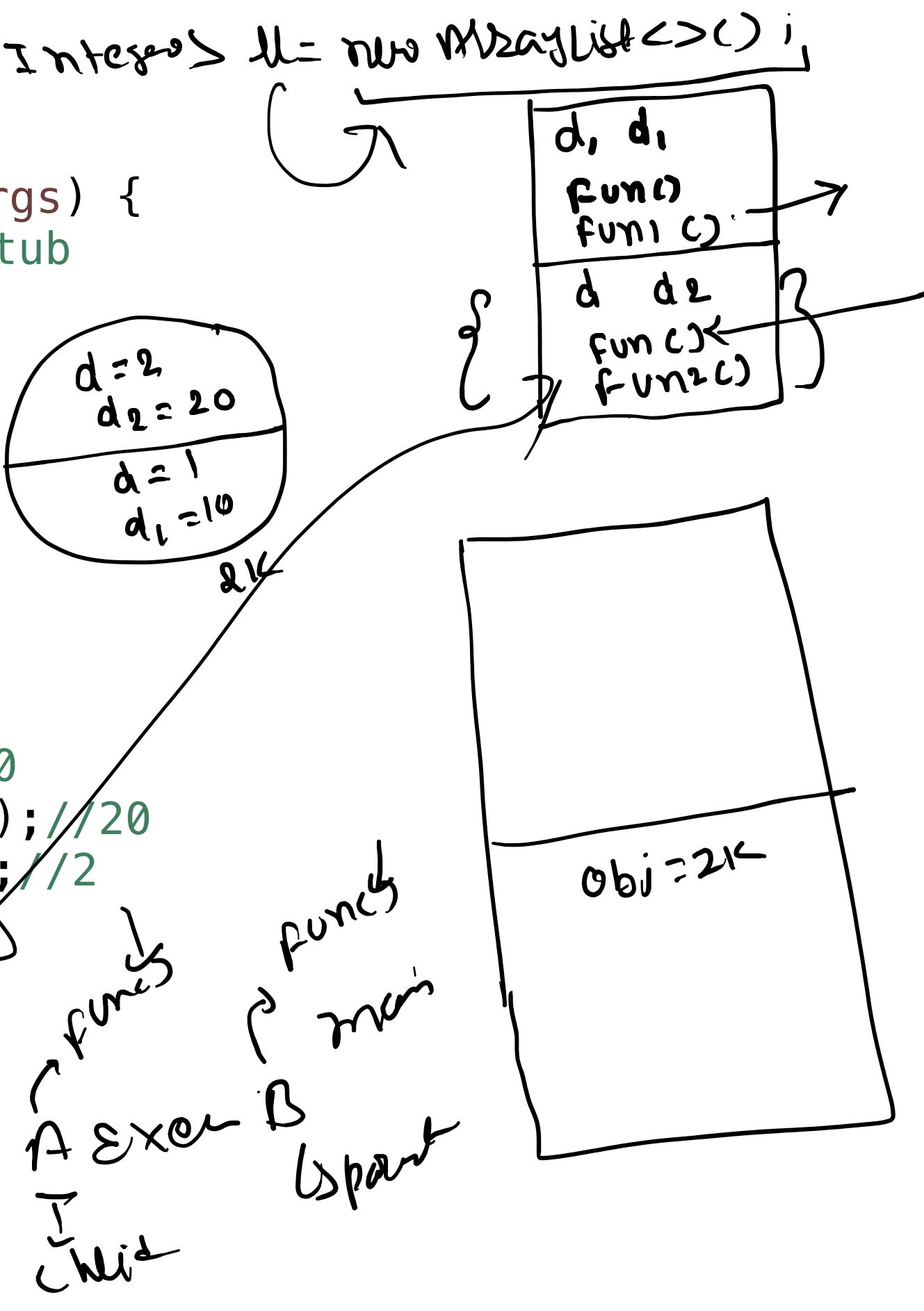
```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    // Case - 1
    P obj = new P();
    System.out.println(obj.d);
    System.out.println(obj.d1);
    obj.fun();
    obj.fun1();
    // Case - 2
    P obj = new C();
}
```



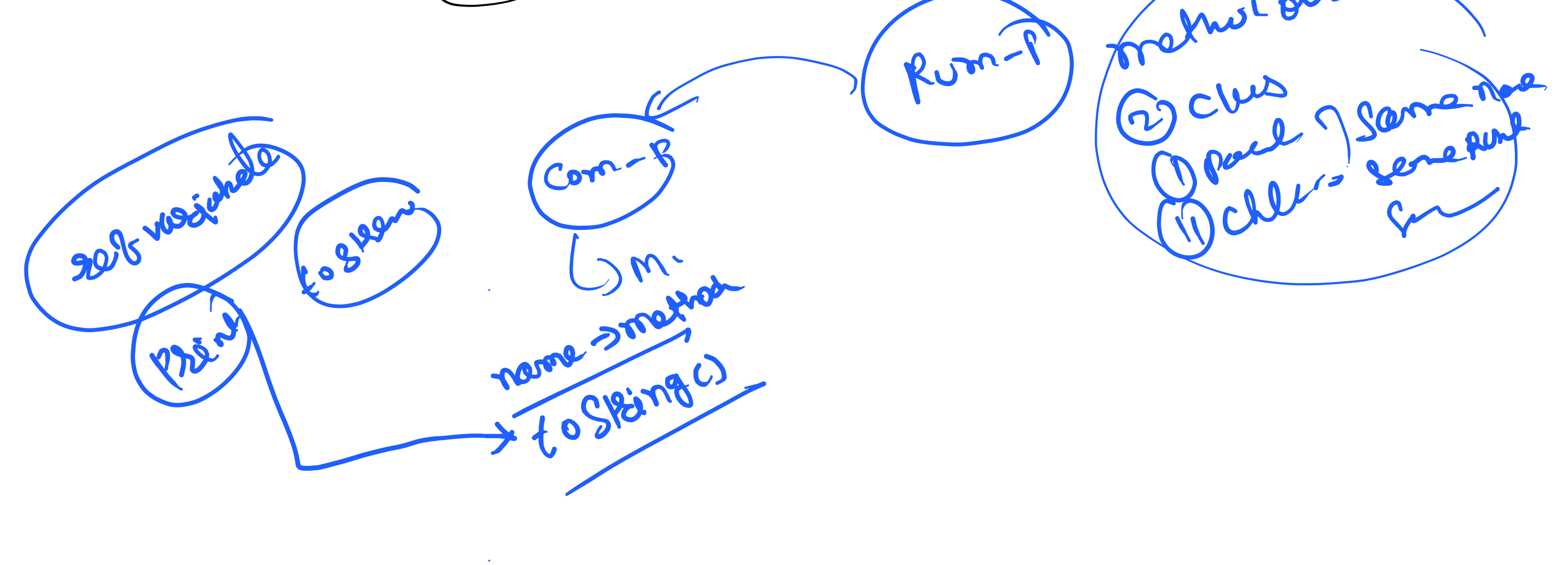
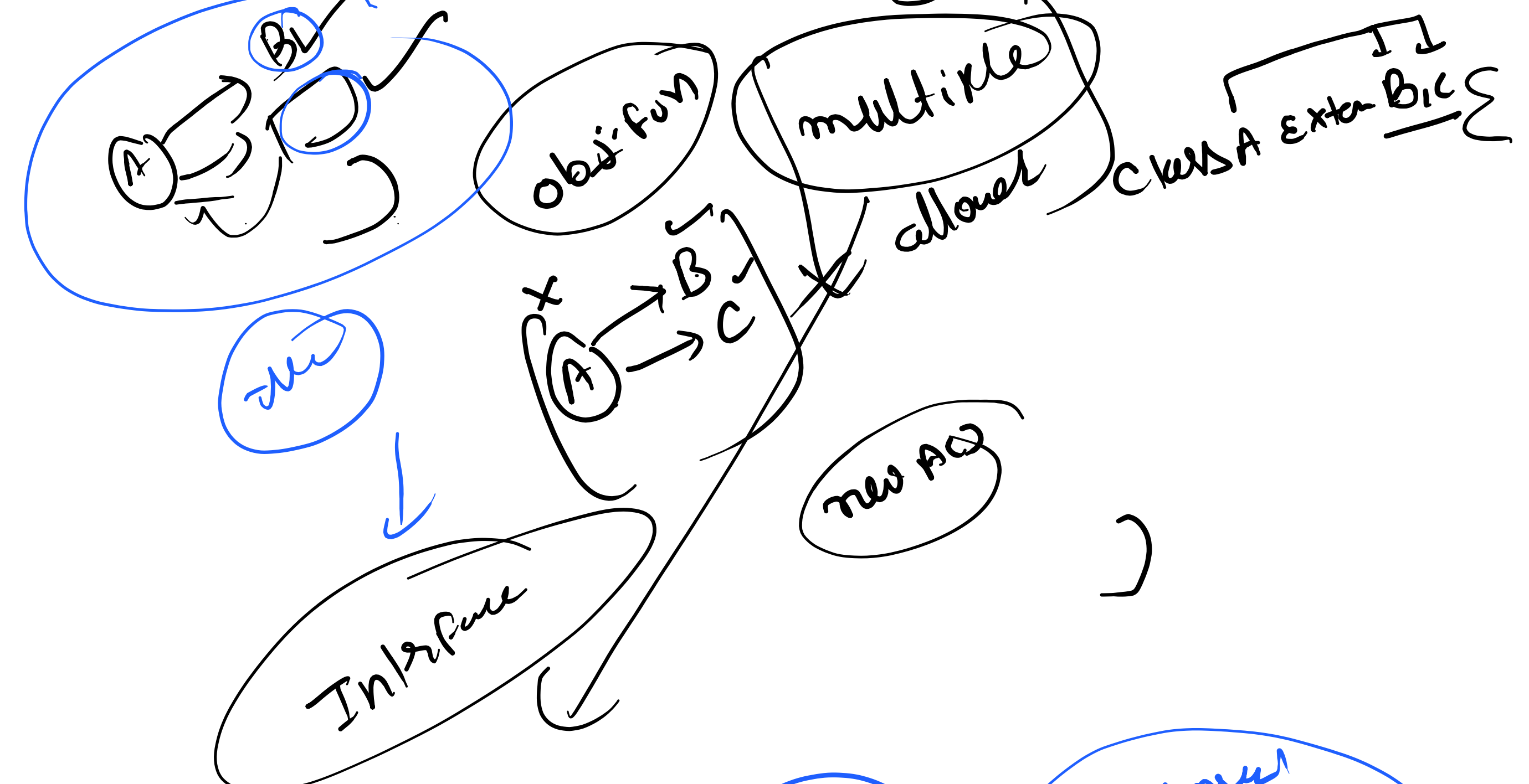
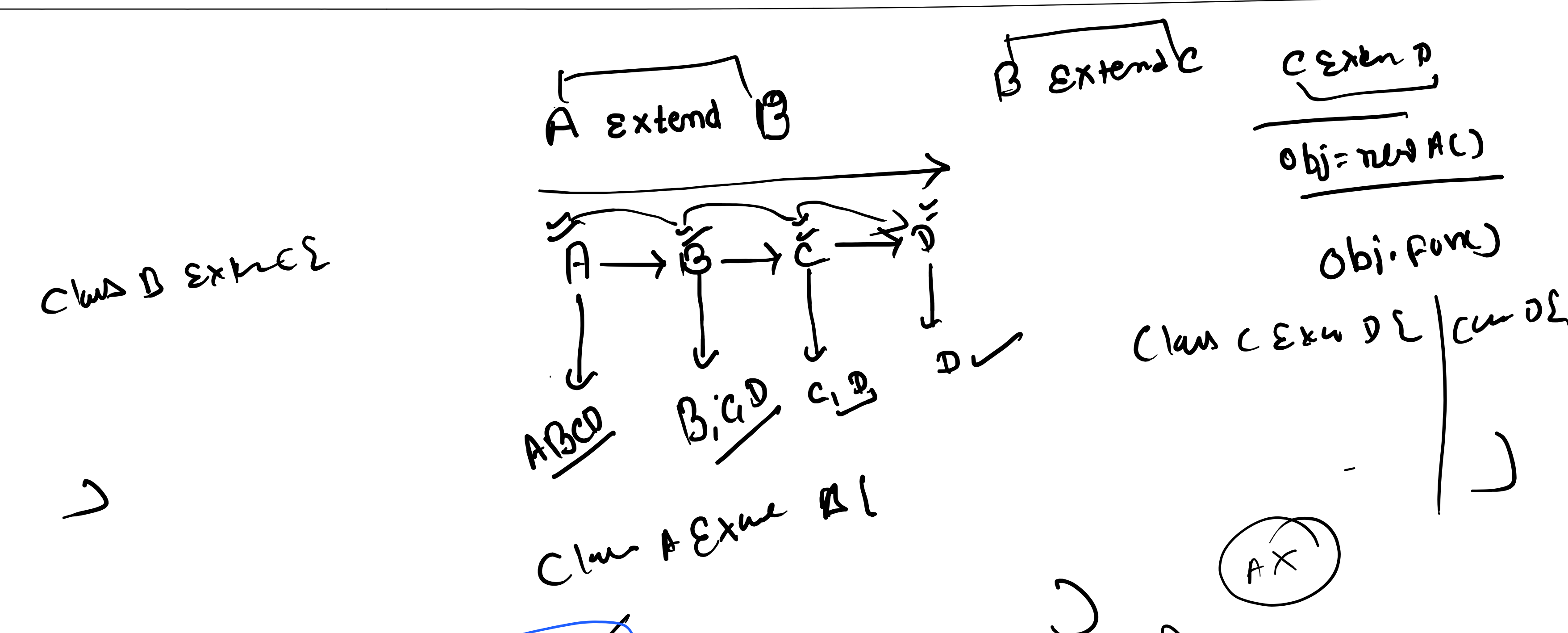
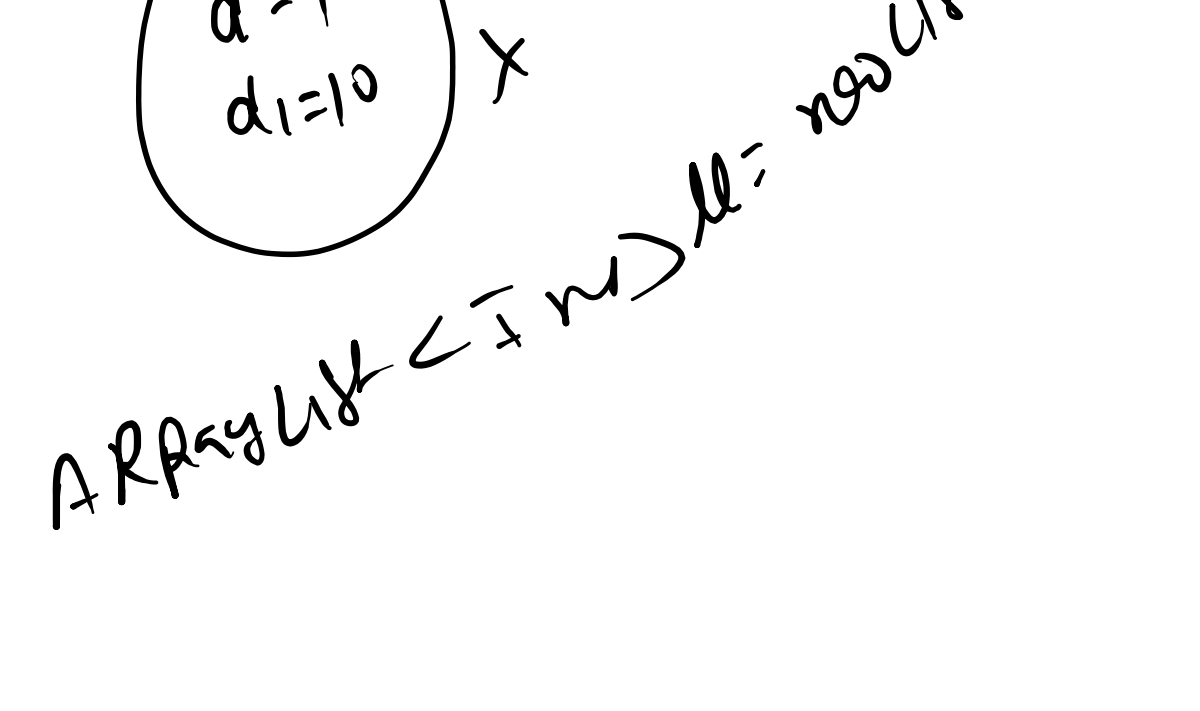
```
public class Client {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        // Case - 1
        P obj = new P();
        System.out.println(obj.d);
        System.out.println(obj.d1);
        obj.fun();
        obj.fun1();
        // Case - 2
        P obj = new C();
        System.out.println(obj.d);
        System.out.println(obj.d1);
        System.out.println(obj.d2);
    }
}
```



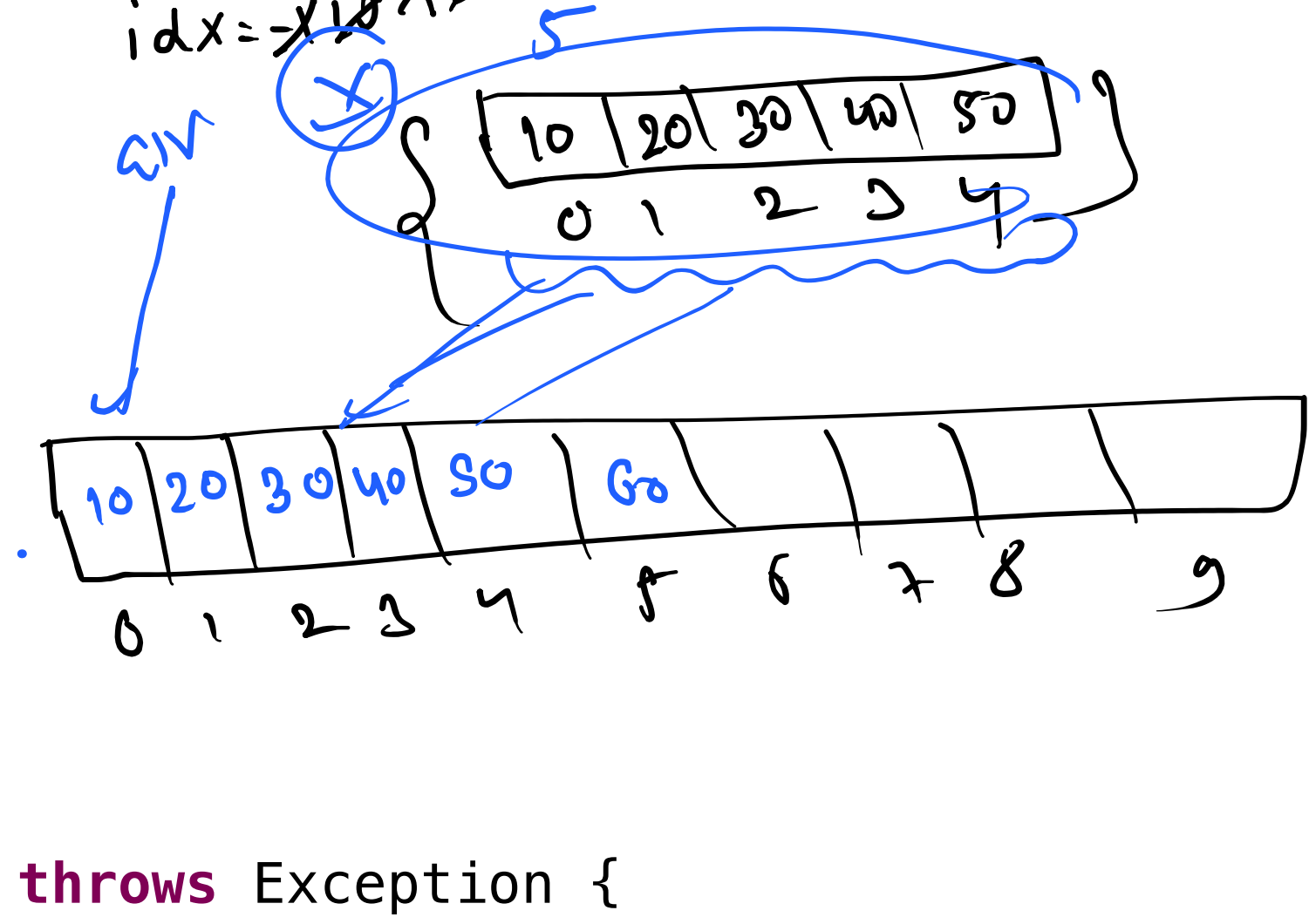
```
public class Client {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        // Case - 1
        P obj = new P();
        System.out.println(obj.d);
        System.out.println(obj.d1);
        obj.fun();
        obj.fun1();
        // Case - 2
        P obj = new C();
        System.out.println(obj.d);
        System.out.println(obj.d1);
        System.out.println(obj.d2);
        obj.fun1();
        ((C) obj).fun2();
    }
}
```



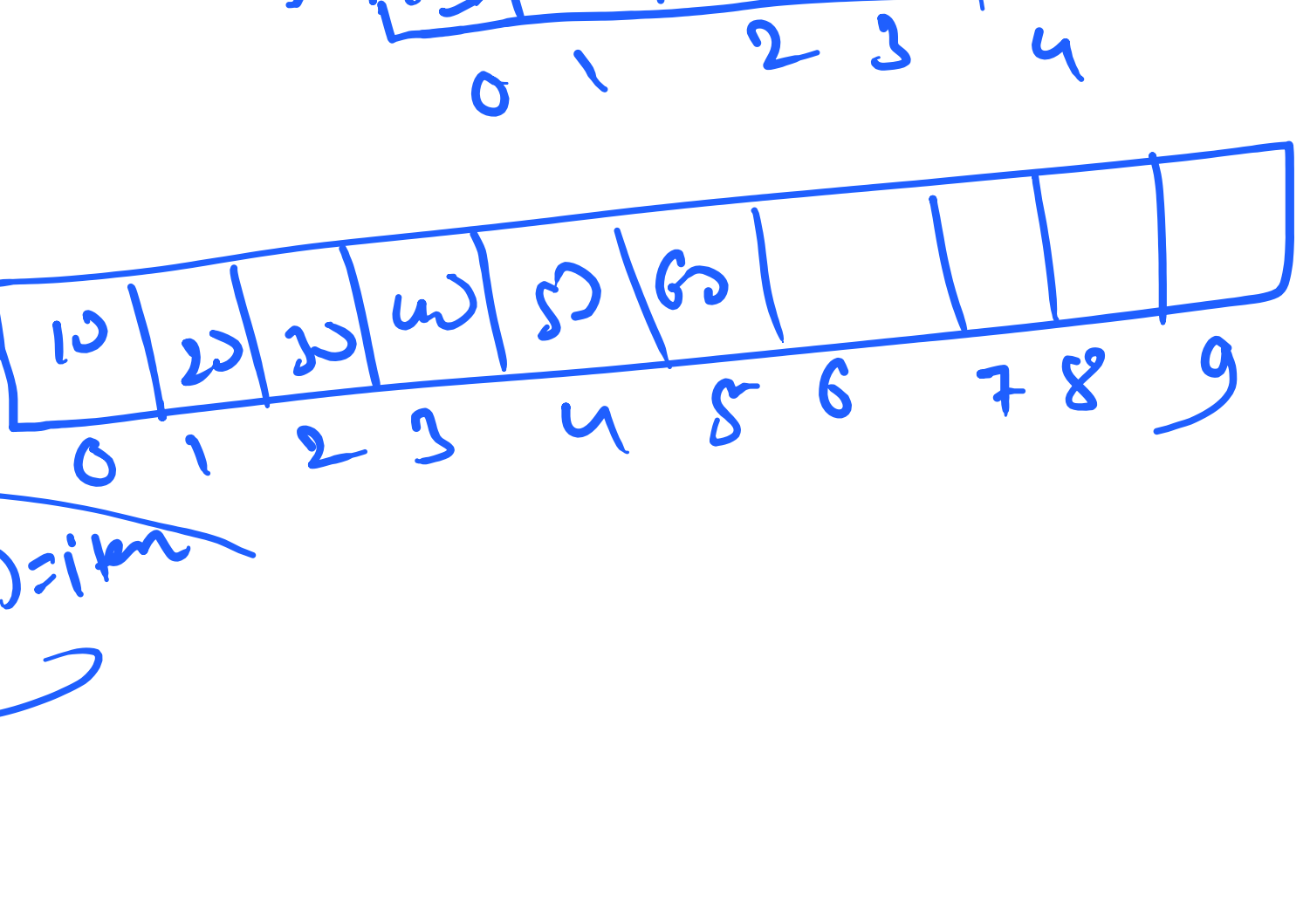
```
public class Client {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        // Case - 3
        C obj = new P();
        System.out.println(obj.d);
        System.out.println(obj.d1);
        System.out.println(obj.d2);
    }
}
```



```
public class DynamicStack extends Stack {
    @Override
    public void push(int item) {
    }
    public static void main(String[] args) {
        DynamicStack ds = new DynamicStack();
        ds.push(10);
        ds.push(20);
        ds.push(30);
        ds.push(40);
        ds.push(50);
        ds.push(60);
    }
}
```



```
public class DynamicStack extends Stack {
    @Override
    public void push(int item) {
        if (isfull()) {
            int[] new_arr = new int[2 * arr.length];
            for (int i = 0; i < arr.length; i++) {
                new_arr[i] = arr[i];
            }
            arr = new_arr;
        }
        // normal odd way se add krenge data
        arr[++idx] = item;
    }
}
```



```
public class DynamicQueue extends Queue {
    @Override
    public void Enqueue(int item) throws Exception {
        if (isfull()) {
            super.Enqueue(item);
        }
    }
    public static void main(String[] args) throws Exception {
        DynamicQueue dq = new DynamicQueue();
        dq.Enqueue(10);
        dq.Enqueue(20);
        dq.Enqueue(30);
        dq.Enqueue(40);
        System.out.println(dq.Dequeue());
        System.out.println(dq.Dequeue());
        dq.Enqueue(50);
        dq.Enqueue(60);
        dq.Enqueue(70);
        dq.Enqueue(80);
        dq.Enqueue(90);
        dq.Display();
    }
}
```

