

TEST 2 REVIEW
FRIDAY NOVEMBER 15

@Test

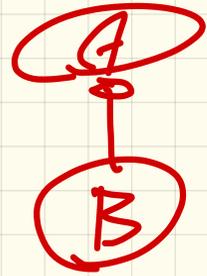
public void test() {

[A obj = new B(...);]

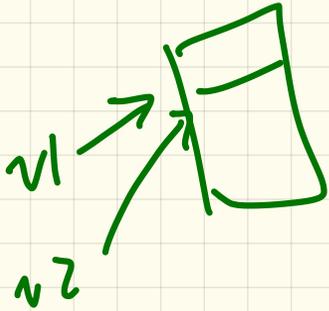
class A { --- }

class B { --- }
extra A.

}



aggregation



1. allow sharing
2. aliasing possible

Composition

1. sharing not allowed
2. aliasing is absent.

```

class Person {
    int age;
}

```

```

PersonCollector pc1 = new PersonCollector();
PersonCollector pc2 = new PersonCollector();
Person p1 = new Person(22);
pc1.addPerson(p1); pc2.addPerson(p1);

```

```

pc1.ps[0] == pc2.ps[0]

```

```

class PersonCollector {
    Person[] ps;
    int nop;
}

```

```

PersonCollector() {
    this.ps = new Person[10];
    this.nop = 0;
}

```

```

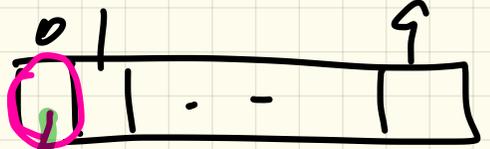
-> void addPerson(Person p) {
    pc1 ps ps[nop] = p;
    pc1 ps nop++;
    pc2 ps ps[0] = p;
    pc2 ps ps[0] = p;
}

```

aggregation -

PC1

PantCollatz	
nop	0
pt	

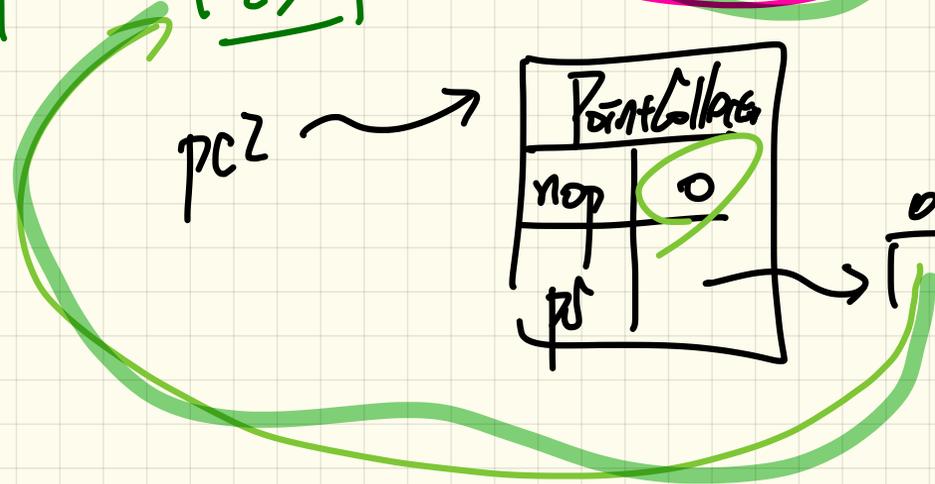
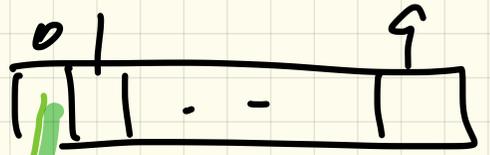


PI

Reson
23

PC2

PantCollatz	
nop	0
pt	



wid addPerson (Person^{p1} p) {

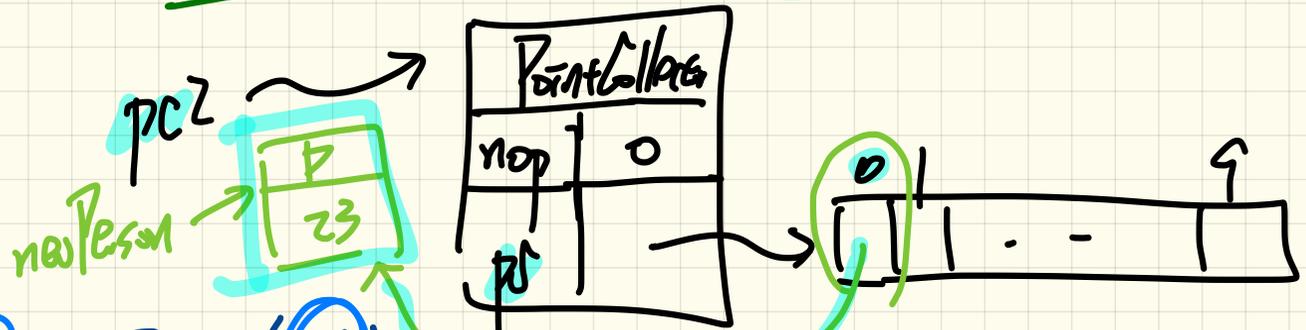
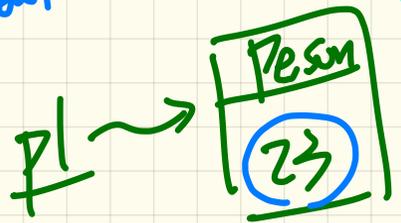
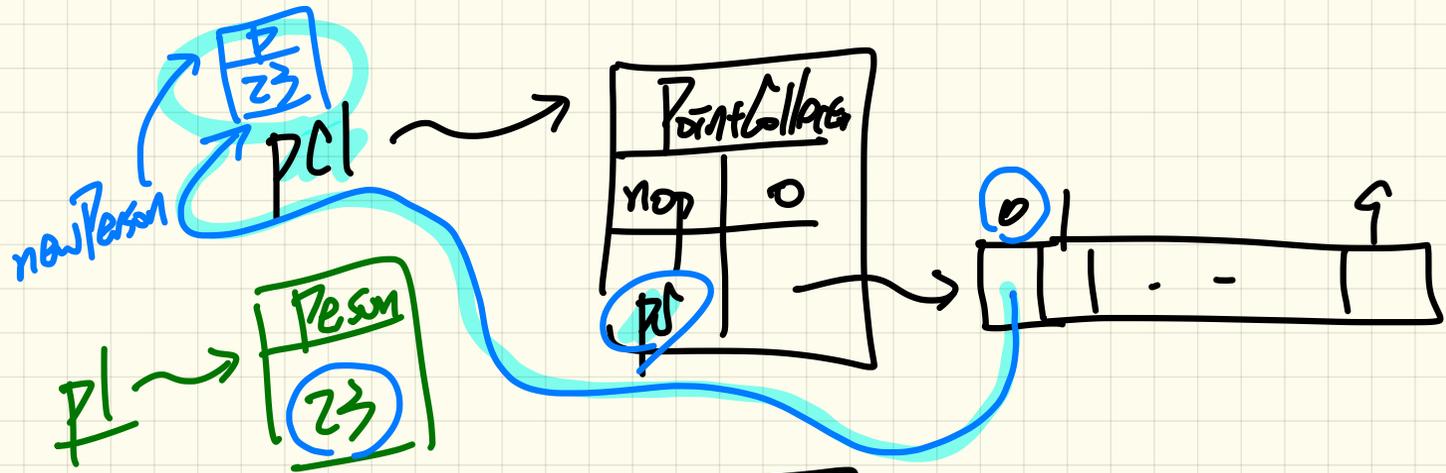
→ Person newPerson = new Person (p.age);

pc2 pc1 ~~this~~. ps [no p] = newPerson;

pc1 ~~this~~. no p ++;

pc2 pc1. ps [no p] = newPerson;

pc2. ps [no p] = newPerson;



```

PC1 addPerson (P);
PC2 addPerson (P);

```

$PC1.ps[0] == PC2.ps[0]$

(F)

```
class PersonCollector {
```

```
    Person[] ps;  
    int nop;
```

```
    PersonCollector (PersonCollector other) {
```

~~this.ps = new Person[~~ps~~];~~
other.ps.
length

PCZ

~~this.nop = other.nop;~~

PC1

```
    for (int i=0; i < this.nop; i++)
```

works for
aggregation.

```
        this.ps[i] = other.ps[i];
```

```
    }  
    this.ps[i] = np;
```

V2

```
        Person np = new Person(  
            other.ps[i].age);
```

V3.

this.ps[i] = new Person(other.ps[i].age);

(V4)

✓ this.ps[i] = new Person(other.ps[i]);

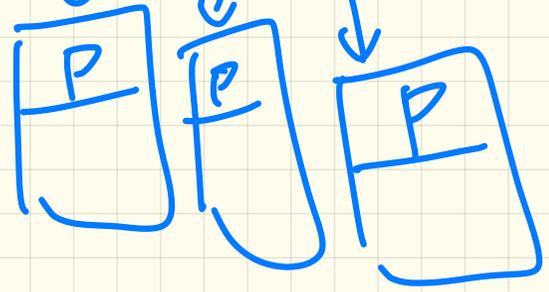
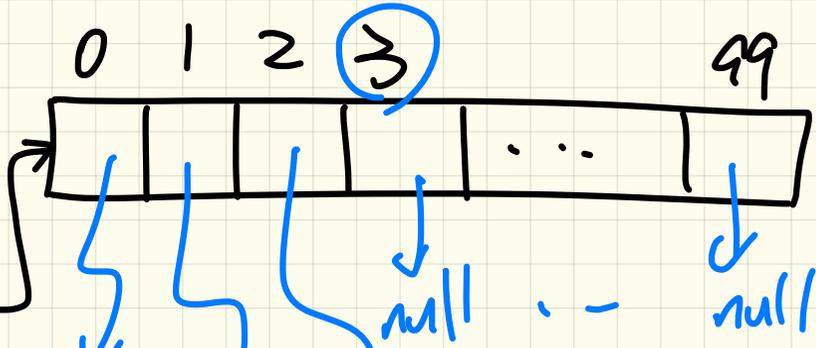
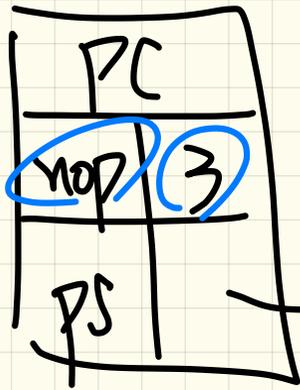
class Person {

Person(Person p) { this.age =
p.age; }

}

VS.

```
PersonCollector (PersonCollector other) {  
    this.nop = other.nop;  
    this.ps = new Person[other.ps.len];  
    for (int i = 0; i < this.nop; i++) {  
        this.addPerson (other.ps[i]);  
    }  
}  
  
void addPerson (Person p) {  
    this.ps[i] = p;  
    this.nop++;  
} →  
this.ps[nop] =  
    new Person(p);  
}
```



nop →
PS.length 100

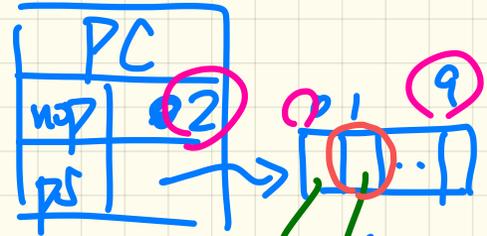
PC pc1 = new PC();

pc1.ps[0] == pc2.ps[0]

pc1.ps[i] == pc2.ps[i]

pc1.addPerson(new Person(23));

pc1.addPerson(new Person(46));



PC pc2 = new PC(pc1);

pc1 →

pc2 this
~~pc2~~.ps[i] = ~~other~~.ps[i]

other

pc2 →

