

Forslag til oppgaver (sett opp på tavlen eller print ut):

1. True eller False. Diskuter i grupper på 2-3 hvert av utsagnene i følgende kode:

```
class A {}
class B extends A {}
class C extends A {}
class D extends C {}

class TrueOrFalse {
    public static void main(String[] args) {
        A a = new A();
        B b = new B();
        C c = new C();
        D d = new D();
        A e = new D();
        A f = new B();

        System.out.println("a " + (b instanceof A));
        System.out.println("b " + (c instanceof D));
        System.out.println("c " + (d instanceof A));
        System.out.println("d " + (e instanceof C));
        System.out.println("e " + (f instanceof C));
    }
}
```

2. Gitt: EVENTUELT HA DENNE SOM GANG

```
class A {}
class B extends A {}
```

- a. Tegn datastrukturtegningen av dette, med de 3 ulike peker-kombinasjoner som går an å lage.
- b. Hvorfor finnes det ikke 4 ulike type pekere? Hva er problemet å med den siste?

Parprogrammering - Parkeringshus

Et parkeringshus har ett visst antall parkeringsplasser. Både motorsykler og biler skal kunne parkere på en parkeringsplass. En sjåfør er en person som må ha fylt 18 år. En motorsykkel kan holde på én person, som er en sjåfør, mens en bil skal kunne holde på alt fra 2 til 5 personer (selvfølgelig spesifisert i det bilen lages), derav en av de må være sjåfør.

1. Tegn opp hvordan dette vil se ut.
2. Med tanke på hva vi har lært om arv, skriv et java-program som implementere dette, dere vi gjenbraker mest mulig kode.
3. Lag unntakshåndtering til parkeringshuset ditt, for å håndtere tilfeller der man prøver å sette inn et kjøretøy når det er fullt, eller på en ugyldig plass.
 - Hint: `ArrayIndexOutOfBoundsException`