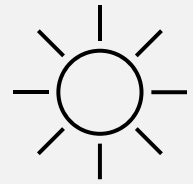
GRUPPE 8 IN1010 – UKE 14

Sivert Fjeldstad Madsen

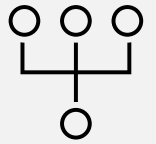
I DAG

- Rekursjon
- Det trivielle tilfellet / Basissteget
- Rekursive kall
- Oppgaver



REKURSJON

- En rekursiv metode er en metode som **kaller på seg selv**
- For hver gang metoden kaller på seg selv må den løse en *enklere* del av problemet
- Til slutt ender man opp med et trivielt steg som avslutter rekursjonen
- Alle problemer som kan løses rekursivt kan også løses iterativt (uten rekursjon)



TELL TIL TI – FOR-LØKKE

```
int tellTil = 10;


for (int i = 0; i < tellTil; i++) {
      System.out.println(i);
}
```

TELL TIL TI – REKURSIVT

```
void tellOpp(int tall) {
	if (tall == 0) {
		return;
	} else {
		tellOpp(tall - 1);
		System.out.println(tall);
	}
}
```

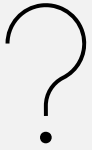
```
	tellOpp(10);
```

DET TRIVIELLE TILFELLET / BASISSTEGET

- Enhver rekursiv metode må ha en tilstand hvor den slutter å gjøre nye rekursive kall

- Dette kaller vi ofte for **det trivielle tilfellet**
  - Hvis vi tenker induksjon tilsvarer dette **basissteget**
- Hvis vi glemmer å håndtere det trivielle tilfellet vil vi kunne ende opp med å gjøre nye rekursive kall i evigheten
- Pass på å håndtere eventuelle edge cases!

HVA ER GALT HER?



```
int opphoeyd(int base, int eksponent) {
    if (eksponent == 1) {
        return 1;
    }
    return base * opphoeyd(int base, int eksponent – 1);
}
```

Må være 0!

REKURSIVE KALL

- Hver gang vi kaller på den rekursive metoden inne i seg selv må vi passe på at problemet har blitt enklere
  - Vi kan tenke på det som at vi må gå ett steg nærmere det trivielle tilfellet
- Glemmer vi dette havner vi fort igjen i en evig løkke
- Merk også at en rekursiv metode gjerne kan gjøre **mer enn ett** nytt rekursivt kall i et rekursjonssteg
- Hvis vi vil at problemet skal løses «nedenfra og opp», gjør vi rekursjonskallet først i metoden
- Hvis problemet skal løses «ovenfra og ned» gjør vi rekursjonskallet til sist

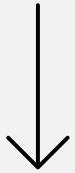
TELL FRA TI – «NEDENFRA OG OPP»

```
void tellOpp(int tall) {
      if (tall == 0) {
            return;
      } else {
            tellOpp(tall - 1);
            System.out.println(tall);
      }
}
```

```
      tellOpp(10);
```

TELL TIL TI – «OVENFRA OG NED»

```
void tellOpp(int tall) {
      if (tall == 0) {
            return;
      } else {
            System.out.println(tall);
            tellOpp(tall - 1);
      }
}
```



```
      tellOpp(10);
```

OPPSUMMERING

- Rekursjon kan fort bli litt forvirrende
- Husk å definere de to komponentene:
  - Basissteget/det trivielle tilfellet
  - Det eller de rekursive kallene

- Ukas tips:
  - Lambda-uttrykk i Java
    - En «anonym» funksjon som kan gjøre mye rart
    - Gjør det mulig å programmere *funksjonelt*