



IN1010 våren 2020

Onsdag 5. februar

Java:

Typer – variabler – uttrykk

Data i minnet

Stein Gjessing

Institutt for informatikk

- **8 primitive typer** med verdier som bruker plass i minnet:
 - boolean kan greie seg med en bit (0 for false og 1 for true)
men kompilatoren velger å bruke én byte
 - byte
 - char: 2 byte
 - short: 2 byte
 - int: 32 bit (4 byte)
 - long: 64 bit
 - float: 32 bit
 - double: 64 bit
- **Referanse-typer er klassenavn*** (objekt-referanser): Så mange bit som det er i datamaskinens adresserom, 32 (eller 64) bit
 - Referanser til arrayer er typet med typen til innholdet + at det er en array
f.eks. `int []`, `Bil []`

Uttrykk regnes ut til en verdi av en av disse typene.
Uttrykk forekommer i høyresider og i aktuelle parametre.

* Senere skal vi se at det også er interface-navn

Javas uttrykk

- Et uttrykk er et «regnestykke» (engelsk: expression) som regnes ut til en verdi av en type (en av Javas 8 primitive typer + referansetyper)
- Uttrykk finner vi i høyresider (HS) i tilordninger og i aktuelle parametre*
 - Pluss at vi finner boolske uttrykk i løkke- og valgtester
 - I return-setninger (som beregner metodens retur-verdi)

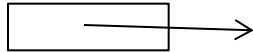
*

Parametrene der metoden er deklarerert kaller vi formell parameter
Parametrene på kallstedet kalles aktuell parameter

Java variabler

- En variabel i Java har et navn, en type og et innhold

Navn: dinTeller



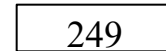
Type: Counter

Navn: minTeller



Type: Counter

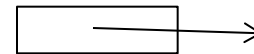
Navn: sum



Type: int

Java's variable får verdi ved tilordning (VS i en tilordning)
og ved parameteroverføring

Navn: alleBiler



Type: Bil []

Scop-regler

En «static»-variabel lever så lenge programmet lever

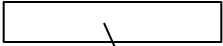
En instansvariabel lever så lenge objektet lever

En variabel deklart i en metode (inkl. parametrene) lever så lenge metodeinstansen lever

En variabel deklart i en blokk lever så lenge programmet eksekverer inne i blokken

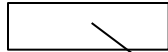
Variabler / objekter

Navn: minTeller



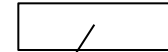
Type: Counter

Navn: hansTeller

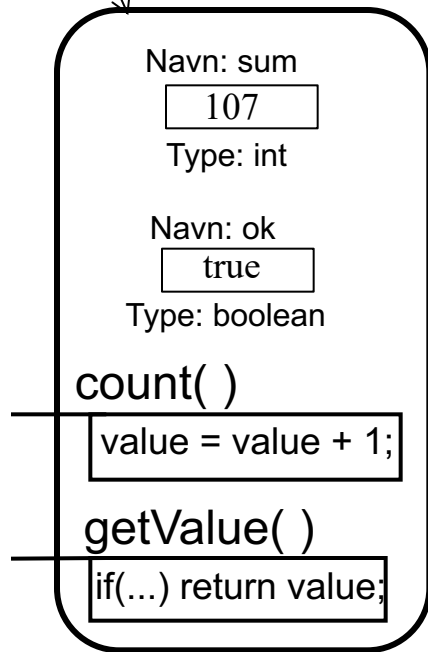


Type: Counter

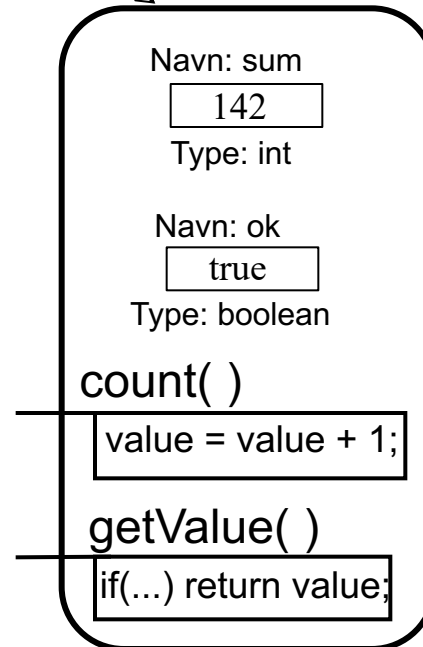
Navn: dinTeller



Type: Counter

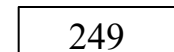


Counter2-objekt



Counter2-objekt

Navn: sum



Type: int

Referanser

Navn: minTeller

53485324

Type: Counter

Navn: hansTeller

59483568

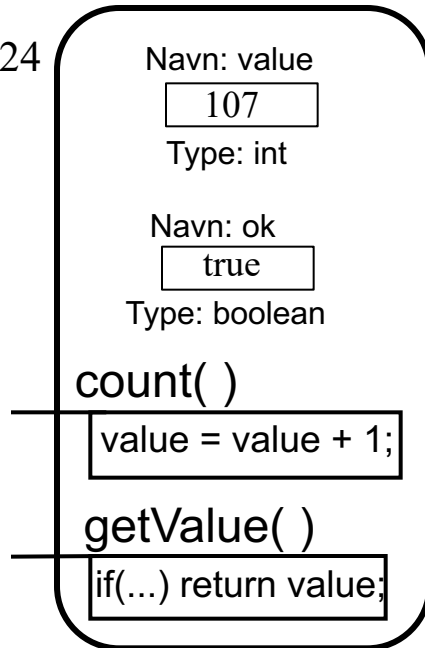
Type: Counter

Navn: dinTeller

59483568

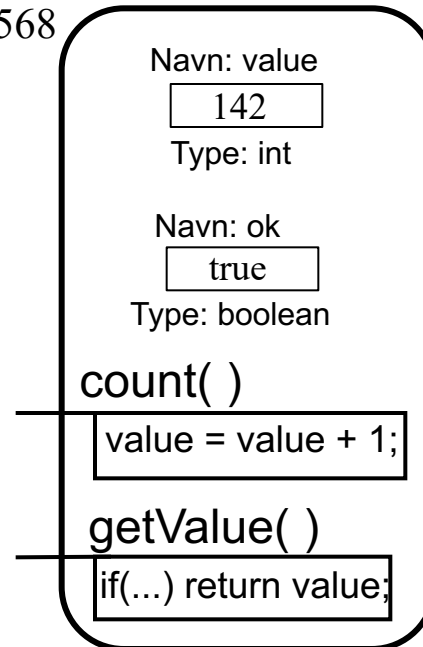
Type: Counter

53485324



Counter2-objekt

59483568



Counter2-objekt

Navn: sum

249

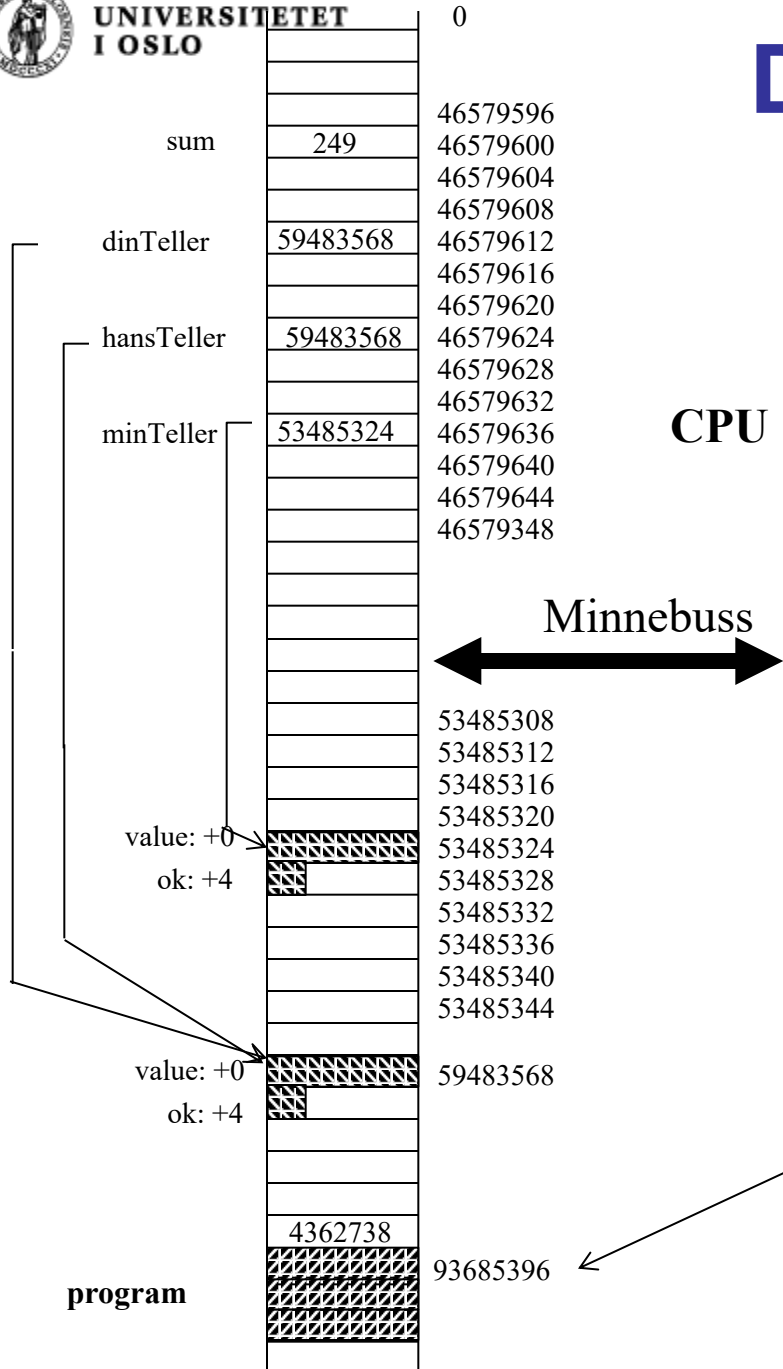
Type: int



Data i minnet

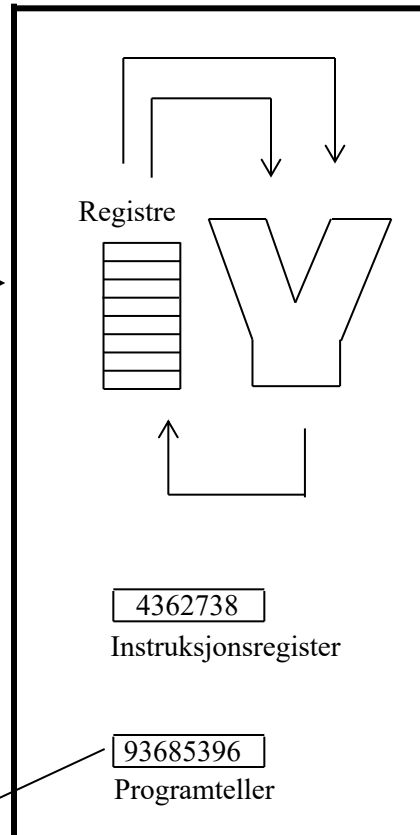
Kompilatoren og kjøretidsystemet setter av
plass til variabler i minnet

Antar 32 bits adresser for referanser



CPU

Minnebuss



sum ligger minnelokasjon 46579600,
46579601, 46579602 og 46579603

dinTeller ligger minnelokasjon 46579612,
46579613, 46579614 og 46579615

hansTeller ligger minnelokasjon 46579624,
46579625, 46579626 og 46579627

minTeller ligger minnelokasjon 46579636,
46579637, 46579638 og 46579639

value ligger i starten av objektet,
mens ok ligger 4 byte ut i objektet

Konstanter i Java

- En konstant i Java deklarereres med «final» og kan ikke endres etter at den er initialisert:

```
class KonstantDemoKlasse {  
    protected final int objektId;  
    KonstantDemoKlasse (int objektId) {  
        this.objektId = objektId;  
    }  
}
```

Helt på siden av dette temaet (men har med scop å gjøre):

«this» er en referanse til objektet som koden er inne i.

Bare når parameteren og instansvariabelen har samme navn brukes «this» i konstruktører!