



PRG1000B Grunnleggende programmering

# Leksjon 3

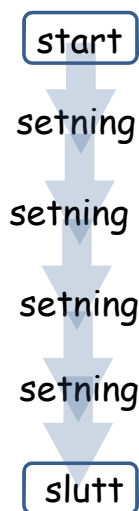
## Kontrollstrukturer

### Del 1

Roy M. Istad, 2017

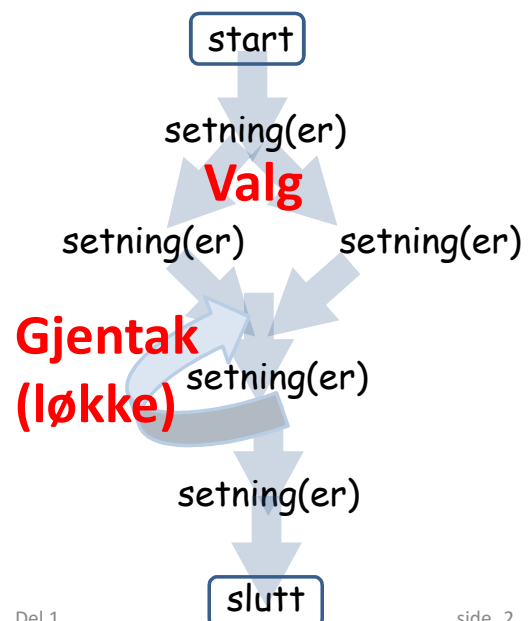
### Utførelse av et program

#### Programflyt så langt:



**Sekvensielt:**  
Alle setninger utføres, den ene etter den andre, og ingen blir hoppet over.

#### Programflyt fra nå av:



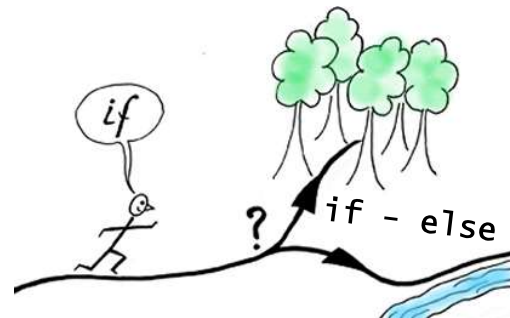
## Valgsetninger

**if**-setning: Valg mellom to alternativ

Enten så er det en mann, eller så er det en kvinne

Enten blir det gitt rabatt, eller så er det fullpris

Enten er man for ung, eller så er alderen høy nok



**switch**-setning: Valg mellom flere alternativ

Til gitt poengsum → Velg prestasjonsomtale

Til gitt dagnummer → Velg ukedag (dagnavn)

Til gitt tegn → Velg kategori  
(vokal, konsonant, siffer, skilletegn, ...)

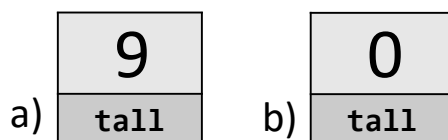


PRG1000B Grunnleggende programmering

Leksjon 3 - Del 1

side 3

## if-setning med else-grein



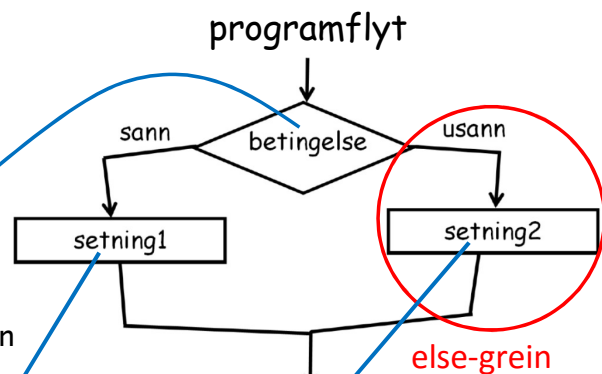
**Eksempel på kode:**

```
int tall = // innlest fra brukeren
if ( tall > 0 )
    showMessageDialog(null, tall + " er positivt!");
else // NB! Da er altså tall <= 0
    showMessageDialog(null, tall + " er ikke positivt!");
```

Utskrift i  
dialogvindu:

a) 9 er positivt!

b) 0 er ikke positivt!



PRG1000B Grunnleggende programmering

Leksjon 3 - Del 1

side 4

## if-setning uten else-grein

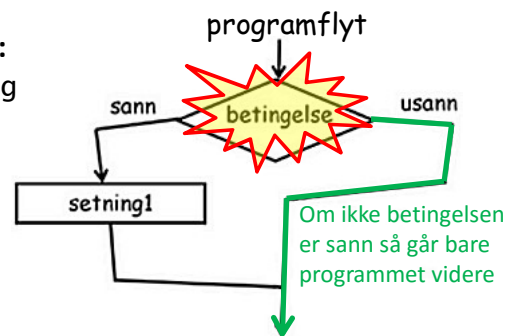
```
int tall = parseInt( showInputDialog("Tall:") );
String melding = "ikke positivt!";
if ( tall > 0 )
    melding = "positivt!";
showMessageDialog(null, "Tallet " + tall + " er " + melding);
```

Alternativ

```
String melding = "positivt!";
if ( tall <= 0 )
    melding = "ikke " + melding;
```

Det må altså ikke være noen alternativ handling, f.eks.:

```
: // kun overtidstimer gir overtidsbetaling
double overBetaling = 0;
if ( overtid > 0 ) {
    // beregn overtidsbetaling
}
: // fortsetter uten noen else-grein
```

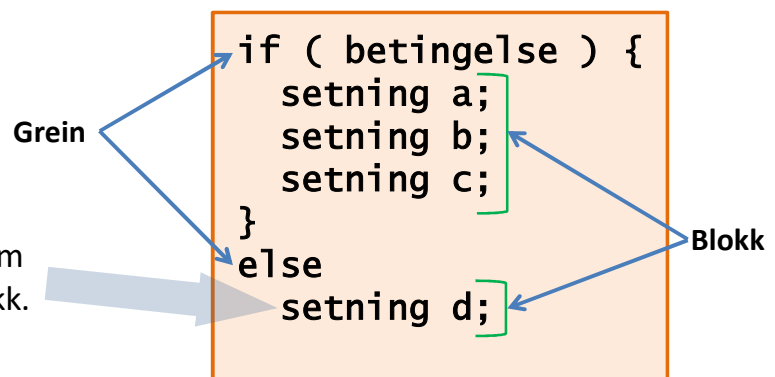


## if-setning: Viktige begrep

- **Betingelse:** Uttrykk, ofte sammenlikning, som beregnes til *sant* (true) eller *usant* (false).
- **Grein:** Tilslagsmulighet i en valgsetning. Består enten av enkelt setning, eller en *blokk*.
- **Blokk:** Setningsliste, en eller flere setninger, avgrenset mellom krøllparenteser { }

**Obs!**

Det er valgfritt med { } dersom det er kun én setning i en blokk.



## Betingelser, logiske uttrykk

- Sammenlikninger:  
tall < 0    alder >= 18    antall == 10    ...    **true/false?**
- Sammenlikningsoperatorer:  

<	<=	>=	>	==	!=
Mindre enn	Mindre enn eller lik	Større enn eller lik	Større enn	Er lik	Ulik, forskjellig fra
- Logiske variabler (**ny** datatype: boolean) kan stå i en betingelse:  
**boolean** myndig = false, student = true;  
 1) if ( myndig )    evt.    2) if ( student )
- Sammensatte uttrykk (jfr. vanlig/matematisk oppsett:  $18 \leq \text{alder} \leq 67$ ):  
 18 <= alder && alder <= 67
- Logiske operatorer:
 

&&		!
og	eller	ikke

PRG1000B Grunnleggende programmering

Leksjon 3 - Del 1

side 7

## Logisk tilordning

- Erstatt følgende (type) if-setning:

```
boolean myndig;  
if ( alder >= 18 )  
    myndig = true;  
else  
    myndig = false;
```

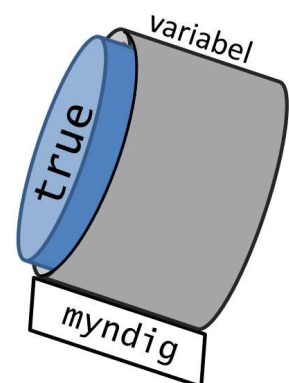
21
alder

- Med følgende (type) tilordning:

```
boolean myndig = (alder >= 18);
```

Obs! Parentesene på høyresiden er overflødige, men bidrar til å gjøre hele tilordningen lettere å lese:

```
boolean jubileum = (ansattÅr == 25);
```



PRG1000B Grunnleggende programmering

Leksjon 3 - Del 1

side 8

## Sammensatte logiske uttrykk

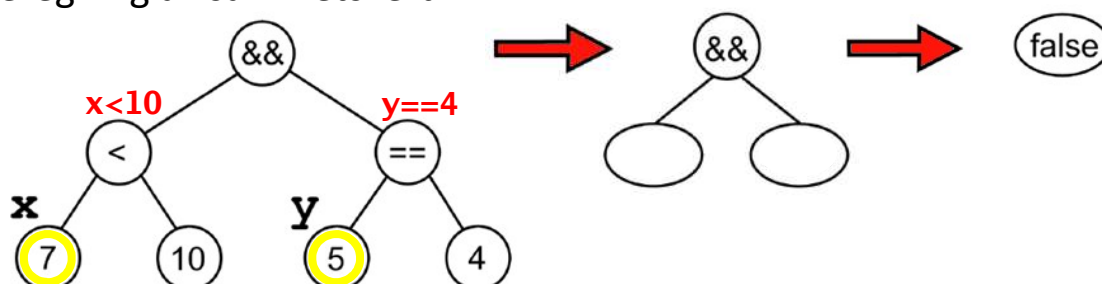
La  $P$  og  $Q$  være to logiske uttrykk (begge kan altså være true/false).

Verdien til sammensetningen av logiske uttrykk kan vises i en «sannhetsverditabell»

P	Q	$P \ \&\& \ Q$	$P \    \ Q$	$!P$

## Beregning av sammensatte uttrykk

- Gitt betingelse: `if ( x<10 && y==4 )`  
→ Evalueres fra venstre
- Beregning av sannhetsverdi:



Her ser vi at  $x$  har verdien 7 og  $y$  har verdien 5

Operator	Beskrivelse
!	ikke
*	multiplikasjon
/	divisjon
%	modul (rest)
+	addisjon
-	subtraksjon
<	mindre enn
<=	mindre enn, eller lik
>=	større enn, eller lik
>	større enn
==	er lik
!=	forskjellig fra
&&	og
	eller

## Operatorprioritet

### Dvs. Rekkefølge for beregning.

- Høyere oppe, høyere prioritet
- Når to operatører har samme styrke, blir de utført fra venstre mot høyre.

F.eks:  $\text{svar} = 7 + 5 * 4 / 3 - 2;$  // svar = 11

- Operatorprioritet er Javas egen måte å sette parenteser på. F.eks. gir disse to uttrykkene samme resultat:

$x + 3 < y * 2 \quad || \quad z > 4 \quad \&\& \quad x == z$   
 $((x + 3) < (y * 2)) \quad || \quad ((z > 4) \quad \&\& \quad (x == z))$

- Vi bruker parenteser for å overstyre operatorprioritet i en beregning – eller for å øke lesbarheten av koden.

PRG1000B Grunnleggende programmering

Leksjon 3 - Del 1

side 11

### Eksempel: Antall dager i en måned.

```

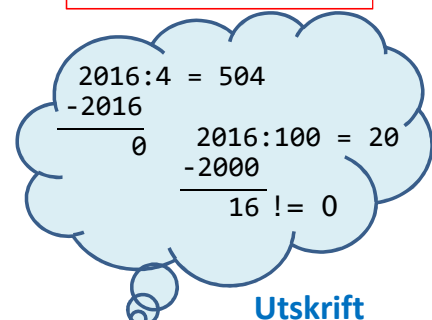
: // innlesing av år og måned (mnd) fra brukeren
int antDager = 31; // Flest måneder har 31 dager
if ( mnd == 4 || mnd == 6 || mnd == 9 || mnd == 11 )
    antDager = 30;
else { // Hva med februar?
    if (mnd == 2) {
        if (( år%4 == 0 && år%100 != 0) || (år%400 == 0))
            antDager = 29; // skuddår
        else
            antDager = 28; // ikke skuddår
    }
}

String utTekst = "Måned " + mnd + " i år " + år + " har " + antDager + " dager.";
:

```

## Nøstet if-setning

Dvs. if-setning inne i if-setning



Utskrift

Måned 2 i år 2016 har 29 dager.

PRG1000B Grunnleggende programmering

Leksjon 3 - Del 1

side 12

**Eksempel:** Karakterbeskrivelse.

Tallkarakter → Bokstavkarakter og tekstlig beskrivelse

```
String karTxt = showInputDialog("Tallkarakter :");
int tallkarakter = parseInt(karTxt);
String karakter; // Tekst som skal få verdi . . .

if ( tallkarakter == 5 )
    karakter = "A - Fremragende";
else // ikke 5 altså . . .
    if ( tallkarakter == 4 )
        karakter = "B - Meget god";
    else
        if ( tallkarakter == 3 )
            karakter = "C - God";
        else // fortsetter slik via 2, 1 og 0 . . .
```

**Nøstet if-setning****Standard oppsett**

Koden krever stor plass  
og kan være vanskelig  
å lese/oppfatte




---

# Slutt på leksjon 3 – Del 1

---