

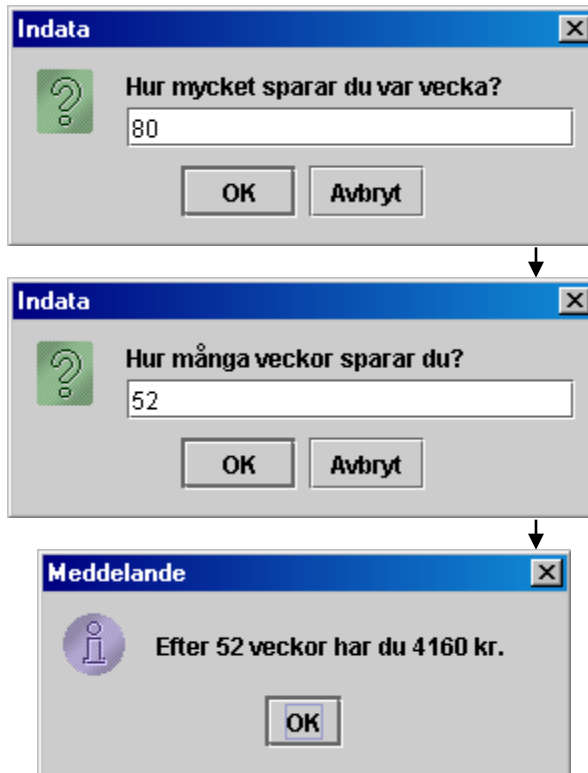
# Föreläsning 8

- Algoritm
- Grundstrukturer i en algoritm
  - Sekvens
  - Selektion
  - Iteration
- boolean
  - jämförelseoperatorer
  - logiska operatorer
- Selektion -
  - if-sats
  - if-else-sats
  - switch-sats

**JF: 4.1 – 4.4**

# Algoritm

## Körresultat



Algoritm - beskrivning över hur man löser problemet i ett antal steg (operationer)

- 1 Deklarera variabler (sparande,veckor,kapital)
- 2 Användaren matar in veckosparande
- 3 Användaren matar in Antalet veckor
- 4 Beräkna kapitalet
- 5 Visa sparresultatet

# Algoritm

En algoritm består av tre olika ”byggstenar” (grundläggande konstruktioner):

- **Sekvens** – en följd av instruktioner
- **Selektion** – val om en sekvens ska exekveras eller ej
- **Iteration** – upprepning (repetition) av en sekvens

De två sista är villkorliga.

- I selektionen avgör ett villkor om en viss sekvens ska utföras eller ej.
- I iterationen avgör ett villkor om sekvensen ska utföras en gång till.

# Algoritm

## Algoritm

- 1 Deklarera variabler
- 2 Inmatning av veckosparande av användaren
- 3 Inmatning av antalet veckor av användaren
- 4 Beräkna kapitalet
- 5 Visa sparresultatet

## Kod

```
int amount, weeks, sum;  
amount = Integer.parseInt( JOptionPane.showInputDialog(  
    "Hur mycket ...?" ) );  
weeks = Integer.parseInt( JOptionPane.showInputDialog(  
    "Hur många ...? " );  
sum = weeks * amount;  
JOptionPane.showMessageDialog( null, "Efter " +  
    weeks + ... );
```

Savings.java

Ovanstående kod beskriven en **sekvens**.

Halvkoden, som var grunden till programmet, kallas för en **algoritm**. Till höger har halvkoden översatts till java.

# Sekvens

Gå och fylla hinken med sand

- 1 Gå med hink och spade till sandhögen
- 2 Fyll hinken med sand
- 3 Gå tillbaka med hink och spade

**Sekvensen** från 1-3 beskriver hur problemet ska lösas. Sekvensen är en följd av steg.

# Iteration

Gå och fylla hinken med sand

Steg 2, **Fyll hinken med sand** kan delas upp i fler instruktioner:

- 1 Gå med hink och spade till sandhögen
- 2.1 Ta sand i spaden
- 2.2 Häll sanden i hinken
- 2.3 Om hinken ej är full så fortsätt med punkt 2.1
- 3 Gå tillbaka med hink och spade

För att hinken ska bli full så måste 2.1-2.3 upprepas ett antal gånger. Detta är s.k. upprepning, dvs. **iteration**.

# Selektion

Gå och fylla hinken med sand

Steg 2.1, **Ta sand i spaden** kan delas upp i fler instruktioner

- 1 Gå med hink och spade till sandhögen
- 2.1.1 Gräv med spaden så det blir sand i spaden
- 2.1.2 Om det är för mycket sand i spaden så
- 2.1.3 häll ut lite sand
- 2.1.4 Om det är för lite sand i spaden så
- 2.1.5 fyll på med lite sand
- 2.2 Häll sanden i hinken
- 2.3 Om hinken ej är full så fortsätt med punkt 2.1.1
- 3 Gå tillbaka med hink och spade

Varje gång som spaden ska tömmas i hinken så måste det vara rimligt mycket sand i spaden. För att det säkerställa detta så utförs 2.1.3 eller 2.1.5 vid behov. Alternativt handlingssätt kallas för **selektion**.

# boolean, jämförelseoperatorer

En variabel av typen boolean kan ha värdet true eller false.

## Jämförelseoperatorer

För att skapa ovanstående värden används ofta följande **jämförelseoperatorer**:

### Exempel

**boolean fortunate;**

**==**      lika med

**!=**      inte lika med

**<**        mindre än

**fortunate = (diseases<1);**

**>**        större än

**fortunate = (savings>10000000);**

**<=**      mindre eller lika med

**>=**      större eller lika med

**fortunate = (children>=1);**

**OBS!**    Parenteserna är ej nödvändiga men väldigt bra att ha!



# Logiska operatorer

Jämförelseoperatorerna kan kombineras med hjälp av de **logiska operatorerna**:

|    |       |                               |
|----|-------|-------------------------------|
| !  | icke  |                               |
| && | och   |                               |
|    | eller | (Alt Gr - < på tangentbordet) |

## Exempel

**boolean fortunate, odd;**

```
fortunate = (savings>100000000) && (diseases<1);  
odd = (shoes.equals("Sandaler")) || (hat.equals("Cylinder"));  
odd = (nbr%2==1);  
odd = !(nbr%2==0);
```

# Formulera villkor

```
int number = Integer.parseInt( JOptionPane.showInputDialog("Ange ett tal") );  
boolean res = villkor;
```

Hur ska **villkor** se ut för att **res** ska bli **true** då

- a number > 67
- b number är 3, 6 eller 7
- c number är i intervallet 25-50
- d number är delbart med 3
- e number är negativt eller större än 10
- f number är ej delbart med 4
- g entalssiffran i number är 7
- h number tillhör något av intervallen 1-4 eller 7-9
- i hundratalssiffran i number är 7

Conditions.java

# Utökad Savings.java

Vi bygger ut programmet Sparande.java på följande sätt:

Innan kapitalet på kontot skrivs ut så delar vi ut bonus till duktiga sparare. Detta sker enligt följande regler:

- Om sparkapitalet är minst 5000 kr så ska spararen ges berömmet "Du är en duktig sparare!" + 100 kr i bonus.
- Annars om sparkapitalet är minst 2500 kr så ska spararen ges samma beröm som tidigare + 50 kr i bonus.
- Annars ges spararen bister kritik över bristande sparförmåga: "Du är ej bra på att spara!"

Därefter meddelas sparresultatet ut.

Savings.java

# Selektion

En selektion innebär att en eller flera satser med kod endast utförs under vissa betingelser. Java innehåller två strukturer för att hjälpa programmeraren med detta:

- **if**-satsen
- **switch**-satsen

Dessutom finns villkorsoperatoren **?**. Kod som använder villkorsoperatoren är ganska svår att läsa och därför ska ni undvika att använda den.

# if-satsen (1)

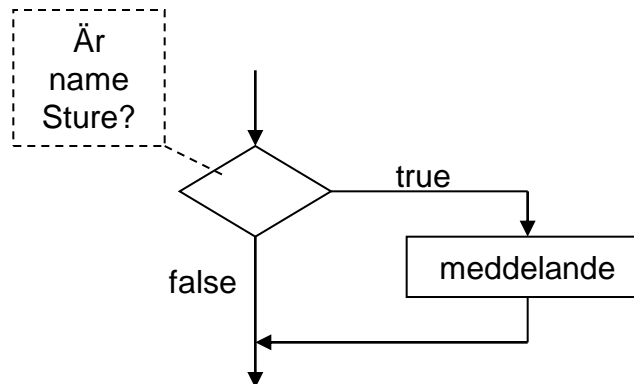
## Gratulation

En person anger sitt namn  
Om personen har namnsdag så  
gratulera på namnsdagen

lf1.java

if( villkor )  
sats;

```
name = JOptionPane.showInputDialog( "Ange ditt namn" );  
if( name.equals( "Sture" ) )  
    JOptionPane.showMessageDialog( "Grattis på namnsdagen!" );
```



# if-satsen (2)

## Inköp av tapeter och lim

Totala priset = antal liter lim \* 97 kr +  
                  antal rullar tapet \* 240 kr +  
                  antal tapetknivar \* 45 kr

Om totala inköpssumman är minst 2000 kr så

    Beräkna 4 % rabatt

    Beräkna vad kunden ska betala

    Meddela kunden rabattens storlek

Ange för kunden vad han ska betala

If2.java

```
if( villkor ) {  
    sats 1;  
    :  
    sats n;  
}
```

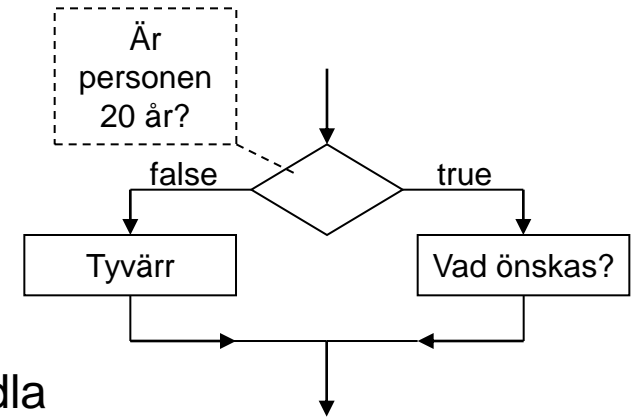
```
total = glue * 97 + wallpaper * 240 + knives * 45;  
if( total >= 2000 ) {  
    discount = 0.04 * total;  
    total = total - discount;  
    message = "Rabatt: " + discount + " kr, " );  
}  
message = message + "Att betala: " + total + " kr";  
JOptionPane.showMessageDialog( null, message );
```

# if-satsen (3)

## Försäljning på Systembolaget

En person anger sin ålder  
Om personen är minst 20 år så  
    Fråga henne vad som önskas  
Annars

    Meddela henne att hon tyvärr ej får handla  
och att hon är välkomment tillbaka om x år.



if(villkor)  
    sats/block  
else  
    sats / block

```
ageStr = JOptionPane.showInputDialog( "Ange ålder" );  
age = Integer.parseInt( age );  
if( age >= 20 )  
    JOptionPane.showMessageDialog( null, "Vad önskas?" );  
else {  
    JOptionPane.showMessageDialog( null,  
        "Ni är för ung för att handla!\n Välkommen tillbaka om" +  
        ( 20 - age ) + " år" );  
}
```

If3.java

# if-satsen (4)

## Betyg

Eleven anger uppnådd poäng på skrivningen

Om poängen  $\geq 20$  så

Betyget är VG

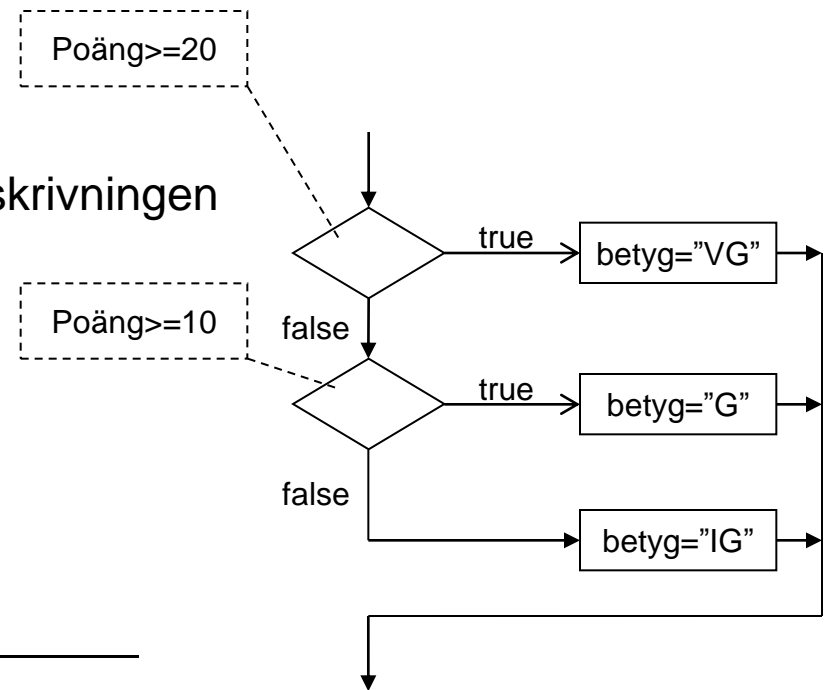
Annars om poängen  $\geq 10$  så

Betyget är G

Annars

Betyget är IG

Meddela eleven hennes betyg



```
if(villkor1)
    sats/block
else if(villkor2)
    sats/block
else
    sats/block
```

```
pointsStr = JOptionPane.showInputDialog( "Ange poäng" );
points = Integer.parseInt( pointsStr );
if( points >= 20 )
    grade = "VG";
else if( points >= 10 )
    grade = "G";
else
    grade = "IG";
JOptionPane.showMessageDialog( "Betyg: " + grade );
```

If4.java



# switch-satsen

## Studentens valsituation kl 15.00 en fredag

Tänkbara val:

- 1 Gå på föreläsning
- 2 Gå hem och läsa kurslitteratur
- 3 Gå till datorsal och jobba

Ange ditt alternativ:

Om du valde 1 så

Meddela "Ett utmärkt val!"

Om du valde 2 så

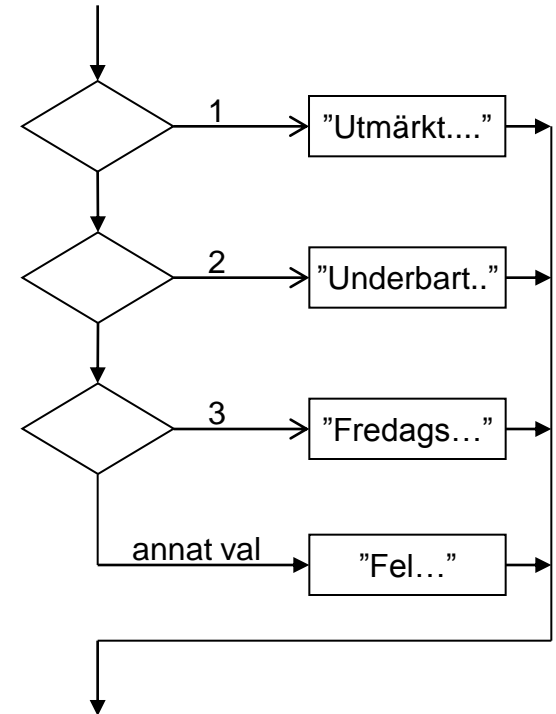
Meddela "Underbart att läsa!"

Om du valde 3 så

Meddela "Fredagsfavoriten!"

Annars

Meddela "Du råkade trycka på en felaktig tangent!"



Konstruera programmet med hjälp av switch-satsen.

Hur ser programmet ut med if-else?

# switch-satsen

```
String menu = "Tänkbara val:\n" +  
              "1 Gå på föreläsning\n" +  
              "2 Gå hem och läsa ...\n" +  
              "3 Gå till datorsal och jobba";  
int choice = Integer.parseInt( JOptionPane.showInputDialog( menu ) );  
  
switch(uttryck) {  
    case a: satser;  
           break;  
    case b: satser;  
           break;  
    case n: satser;  
           break;  
    default: satser;  
}  
  
switch(choice) {  
    case 1: message = "Ett utmärkt val!";  
           break;  
    case 2: message = "Underbart ...!";  
           break;  
    case 3: message = "Fredagsfav...";  
           break;  
    default: message = "Du råkade ...";  
}  
JOptionPane.showMessageDialog( null, message );
```

- **uttryck** ska vara heltal eller char.
- efter case ska det följa en **konstant**, t.ex. 3 eller 'b'
- om samma kod ska utföras för flera case så räknar man upp dem med kolon mellan.  
**case 1 : case 2 : case 8 : System.out.println(...); break;**
- glöm ej **break**;

SwitchProgram.java

# Ett program om ålder

## 4 exempel på körresultat

|  |  |
|--|--|
| Ange ditt födelseår: 1988<br><br>Du är 17 i år.<br>Du är omyndig.    | Ange ditt födelseår: 1961<br><br>Du är 44 år.<br>Du är i arbetsför ålder.  |
| Ange ditt födelseår: 1901<br><br>Du är 104 i år.<br>Du är pensionär. | Ange ditt födelseår: 1966<br><br>Du är 39 år.<br>Nästa år fyller du jämnt. GRATIS!!!<br>Du är i arbetsför ålder. |

Användaren ska ange sitt födelseår. Sedan ska programmet:

- Skriva ut personens ålder i år
- Gratulera om personen fyller jämnt nästa år, dvs 10, 20, 30, 40, ...
- Skriva ut om personen är
  - \* Omyndig (0-17 år)
  - \* Arbetsför (18-64 år)
  - \* Pensionär (65 år och uppåt)