

Föreläsning 3

DD1315

Programmeringsteknik

7,5 hp

Innehåll

- Jämförelseoperatorer
- Logiska operatorer
- Villkorssatsen *if*
 - *elif*
 - *else*
- *Slingor*:
 - *while*
 - *for*
 - *break*

Jämförelseoperatorer

- Jämförelseoperatorer är booleska uttryck, vilket innebär att de antingen är sanna (*True*) eller falska (*False*).

Operator	Tolkning
==	Lika med
!=	Skiljt från
<	Mindre än
<=	
>	Större än
>=	

Logiska operatorer

- Villkor kan kombineras med följande operatorer.

Operator	tolkning
<i>villkor1</i> and (&) <i>villkor2</i>	Båda villkoren måste vara uppfyllda
<i>villkor1</i> or () <i>villkor2</i>	Det ena måste vara uppfyllt
not villkor	Invertering

If-satsen

- if-satsen används när ett villkor avgör vad som ska göras och den kod som ska exekveras om villkoret är uppfyllt måste indenteras:

```
if poäng < 5:  
    print ('U')
```

- Vill man bygga på med fler “case” kan man addera *elif* (*else if*) sats:

```
elif poäng < 10:  
    print ('E')
```

- Man kan lägga till *else* för att tala om vad som ska hända om villkoret inte är uppfyllt:

```
if poäng < 5:  
    print ('U')  
else:  
    print ('G')
```

Exempel

```
# If.py
```

```
print ('Adressregister, välj:')  
print ('1 - Sök efter person')  
print ('2 - Lagra ny person')  
print ('3 - Avsluta')
```

```
val = input('Ditt val: ')  
if val == '1':  
    print ('Sök')  
elif val == '2':  
    print ('Lagra')  
elif val == '3':  
    print ('Avsluta')  
else:  
    print ('Felaktigt val')  
print (val)
```

Slingor

- När man vill att vissa satser ska utföras flera gånger använder man en slinga.
- För att visa vilka satser som ska upprepas måste dessa indenteras.
- Man kan välja mellan att
 - ange ett villkor för hur länge satserna ska upprepas (*while*-slinga)
 - tala om hur många gånger satserna ska upprepas (*for*-slinga)

while-slingan

- Så länge som villkoret är uppfyllt utförs satserna:

while val != '3':

Satser som ska upprepas

fortsättning

- Satserna utförs 0, 1 eller flera ggr.
- En *while-slinga* kan avbrytas med *break*:

if val == '3':

break

Exempel

```
import random
korrektSvar = False
while not korrektSvar:

    tal1 = random.randint(0,10)
    tal2 = random.randint(0,10)
    frågerad = str(tal1) + '*' + str(tal2) + ' = '

    svar = input(frågerad)
    svarTal = int(svar)
    if (svarTal == tal1*tal2):
        print('Rätt')
        korrektSvar = True
    else:
        print('Fel. Gör om.')
print ('Slut')
```

for-slingan

- Vet man hur många varv slingan ska gå runt kan man använda *for* istället:

for i in 0,1,2:

#Satser som ska upprepas

fortsättning

- Även här kan *break* användas

Exempel

```
import random
```

```
krona = 0
```

```
klave = 0
```

```
for i in range(100):
```

```
    mynt = random.randint(0,1)
```

```
    if ( mynt == 0 ):
```

```
        krona=krona + 1
```

```
    else:
```

```
        Klave+=1 % alternativt skrivsätt för att öka en variabel med 1
```

```
print ('Antal krona = ', krona)
```

```
print ('Antal klave = ', klave)
```