

Skolan för Datavetenskap och kommunikation

DD1310/DD1314/DA3009 Programmeringsteknik

Föreläsning 1

Kursinfo
 Diagnostiskt prov
 Python-intro:
 print
 variabler
 reserverade ord
 input
 kommentarer
 beräkningar
 datatyper
 if-satser

Vilka läser kursen?

- Samhällsbyggnad
 - Civilingenjör och Lärare
 - Datorlingvistik (SU)
 - Kursnämndsrepresentanter?
- Labbar på V
(Windows)*
*Labbar på CSC
(Ubuntu)*

Lärandemål...

Efter godkänd kurs ska du kunna

- Python's syntax
- god programmeringsstil
- hitta och rätta fel
- ändra färdiga program
- hämta data från fil
- lagra data på fil

...fler lärandemål

- villkor och slingor
- funktioner
- problem -> program
- datastrukturer (t ex listor, klasser)
- enkla grafiska gränssnitt
- granska andras program

VARFÖR?

för att du ska kunna:

- använda programmering för att lösa problem,
- tillämpa problemlösningsmetodiken även inom andra områden än programmering,
- diskutera programutveckling med experter,
- bedöma kommersiella program.

UNDERVISNING

- Entimmesföreläsningar med läsanvisningar.
Kom förberedd för bästa resultat!
- Labbtimmar - handledning
- Övningstimmar - redovisning

VAD SKA DU GÖRA?

Laborationer (LAB1; 1,5hp)

- Fem labbar: program & instuderingsuppgifter

Prov (LAB2; 1,5 hp)

- Skriftligt prov på grunderna

P-uppgift (LAB3; 3p)

- Större individuellt programmeringsprojekt

Webblabbar (endast CLGYM/Datorlingv)

BONUSPOÄNG

Moment	Poäng	
Labb 1-5	0-4/labbar	Skriftligt förhör på instuderingsfrågor: <ul style="list-style-type: none"> • tre rätt ger godkänt • fyra rätt ger 1 bp • fem rätt ger 2 bp Uppfylls kraven i labbdelen? <ul style="list-style-type: none"> • två krav uppfyllda ger godkänt • tre krav uppfyllda ger 1 bp • fyra krav uppfyllda ger 2 bp
Provet	0-4	
Spec	0-8	
Prototyp	0-8	

Se KTH Social!

KURSFÖRBEREDNING

ENKLASTE PROGRAMMET

```
print("Hej")
```

En *sats* som skriver ut Hej på skärmen.

Kommandot heter print (PRINT eller Print fungerar inte)

Texten "Hej" kallas för ett *uttryck*

sats=statement

kommando=command

uttryck=expression

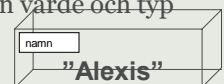
VARIABLER

Variabler - lagrar data i programmet.

Variabelnamn (bokstäver, siffror, _):
antal, vikt2, bioBiljetter,
serie_mördare

Tilldelning: ger variabeln värde och typ

```
namn = "Alexis"
print("Grattis" + namn)
```



tilldelning=assignment

TYPER

Varje variabel har en typ, t ex

namn	innebörd	exempel
int	heltal	5 -239769
float	flyttal	3.14 4e-3
bool	villkorsvärde	True False
str	sträng	"eld" "12"

RESERVERADE ORD

Följande ord är reserverade i Python:

```
and      del      for      is      raise
assert  elif     from     lambda try
break   else     global  not     while
class   except  if      or      return
continue exec  import  pass
def     finally in      print
```

Reserverade ord har betydelse i språket och får inte användas som variabelnamn.

UPPGIFT: VILKA AV FÖLJANDE ÄR OK SOM VARIABELNAMN?

```
anka
and
må
moment22
pass
passfoto
akademiskaHus
17tåget
el-ledning
jord_prover
```

INLÄSNING (PYTHON 3)

•Funktionen input() används vid inläsning av strängar:

```
namn=input("Vad heter du?")
print("Nämen", namn, "då!")
```

•Vill vi ha tal kan vi konvertera med t ex int(input())

```
storlek=int(input("Ge skostorlek: "))
print("Ta "+str(storlek+1)+" i skridskor")
print("så får du plats med sockor också!")
```

Inläsning=user input

Kommentarer

- Alla rader som börjar med # blir kommentarer.

```
# Programmet som ger komplimanger
# Skrivet av Linda Kann 130115
print("Hej, ")
print("Vilken fin klänning,")
print("och vad gott du luktar!")
```

BMI-PROGRAM

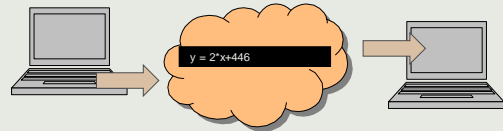
```
# Läser in längd och vikt,
# skriver ut BMI (Body Mass Index)
print("Välkommen till BMI-beräknaren!")
langd=input("Hur lång är du (i meter)? ")
vikt=input("Hur mycket väger du (i kg)? ")
bmi=vikt/(langd*langd)
print("Din bmi är: ", bmi)
```

UPPGIFT: I VILKEN ORDNING SKA SATSERNA STÅ?

```
a.sidorPerDag = sidor/dagar
b.sidor = 63
c.print("Antal sidor per
dag:",sidorPerDag)
d.dagar = 3
```

ALGORITM

1. Läser in indata
2. Gör beräkningar
3. Skriver ut resultatet



Heltalsberäkningar

Operator:	Beskrivning:	Exempel:	Resultat:
*	multiplikation	3*4	12
/	division	53/10 53//10	5
%	modulo	53%10	3
+	addition	10+12	22
-	subtraktion	5-8	-3

heltal=integer

UPPGIFT: HUR KAN MAN ANVÄNDA % FÖR ATT TA REDA PÅ OM ETT TAL ÄR JÄMNT ELLER UDDA?

Flyttalsberäkningar

Operator	Beskrivning	Exempel	Resultat
*	multiplikation	2.0*1.5	3.0
/	division	10.0/8.0	1.25
%	modulo	4.25%4.0	0.25
+	addition	0.3+0.4	0.7
-	subtraktion	1.0-0.1	0.9

flyttal=floating-point number

STRÄNGAR

En *sträng* är en följd av tecken.

Strängar *konkateneras* med +

tex blir "kus" + "lig" strängen "kuslig"

Strängar upprepas med *

tex blir "nä"*3 strängen "nänänä"

Sträng är en *datatyp*. Andra datatyper är *heltal* och *flyttal*.

sträng=string

Konkatenera (slå ihop)=concatenate

Typkonvertering

Funktion	Beskrivning	Exempel:	Blir
<code>float(x)</code>	Konverterar till flyttal	<code>float("3.14")</code>	3.14
<code>int(x)</code>	Konverterar till heltal	<code>int("17")</code>	17
<code>str(x)</code>	Konverterar till en sträng	<code>str(39)</code>	"39"

parameter=argument

Styrstrukturer

I ett program utförs satserna i ordning uppifrån och ner:

```
print("So long")
print("and thanks")
print("for all the fish!")
```

Hur gör man för att hoppa över en sats eller för att upprepa en sats flera gånger?

```
talet = 17
print("Gissa talet - fina priser!")
gissning=int(input("Din gissning:"))
if gissning == talet:
    print("Hurra, du vann!")
    print("Tyvärr är priserna slut.")
else:
    print("Fel svar.")
print("\n--- Välkommen åter ---")
```

if-satsen

```
if villkor:
    block1
elif:
    block2
else:
    block3
```

if-satsen används för val mellan två alternativ, *elif* och *else* kan användas vid behov.

Raderna efter kolon bildar ett *block*: en eller flera satser som är *indenterade* (tabbar i början av raden).