Pythonintroduktion
2023

```
data = np.loadtxt('population.csv', dtype={'names': ('year', 'totpopulation'), 'formats': (np.int, np.float)}, delimiter=',', s


years = data['year']
pop = data['totpopulation']

#medelvärde procentuell ökning 1950-2019
pop_deriv = np.mean(np.diff(pop[-70:]) / pop[-1]) + 1

#medelvärde procentuell ökning 2010-2019
pop_deriv2 = np.mean(np.diff(pop[-10:]) / pop[-1]) + 1


#exponetiell regression (minsta kvadrat)
pop_reg = optimize.curve_fit(lambda x, A, B: A * (B ** x), years, pop, p0=(1,1.01))


#år fram till 2050
futureYears = np.arange(2019,2051)

future_popC = np.empty(32)
future_popB = np.empty(32)
future_popA = pop_reg[0][0] * pop_reg[0][1] ** futureYears

for i in range(len(future_popC)):

    pop_i = pop[-1] * (pop_deriv ** (i))
    future_popC[i] = pop_i

    pop_i2 = pop[-1] * (pop_deriv2 ** (i))
    future_popB[i] = pop_i2

fig, ax = plt.subplots(1,1)
```

Upplägg

Pass 1: Tisdag 22/8 13:15 – 15:00
Pass 2: Torsdag 24/8 13:15 – 15:00

Första timmen:

Varför vi programmerar, Instruktioner, Variabler & Matematik

Första halvtimman:

1. Vad är programmering?
2. Hur man skriver kod
3. Grundläggande matematiska operationer
4. Variabler

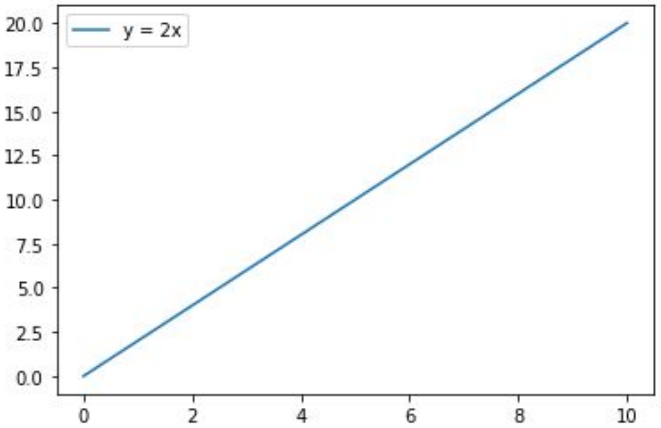
Vad är programmering?

Man vill få datorn att göra som man vill

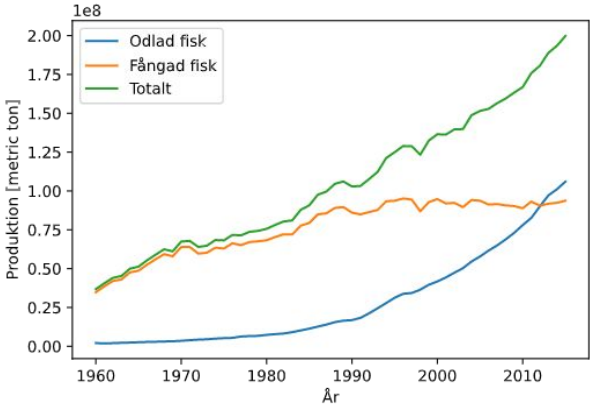
Kan göra:

Långa beräkningar
Visualisering

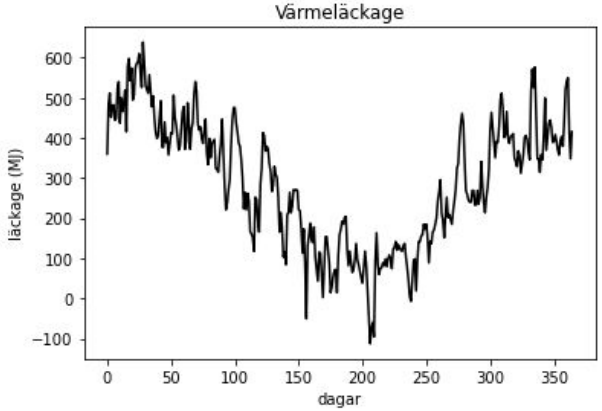
Språk: Python



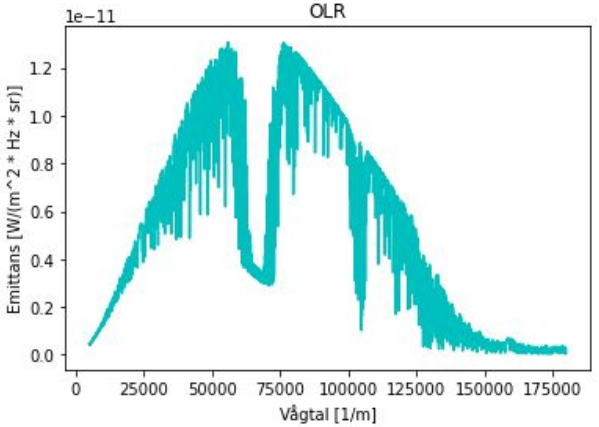
Fiskpopulation
(Linjär algebra)

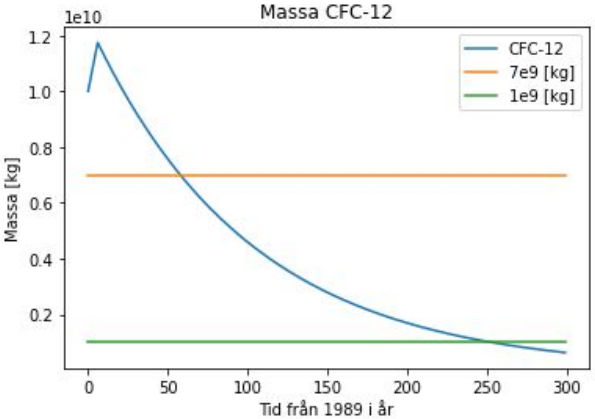


Värmeläckage från hus
(Ordning och oordning i fysikaliska system)

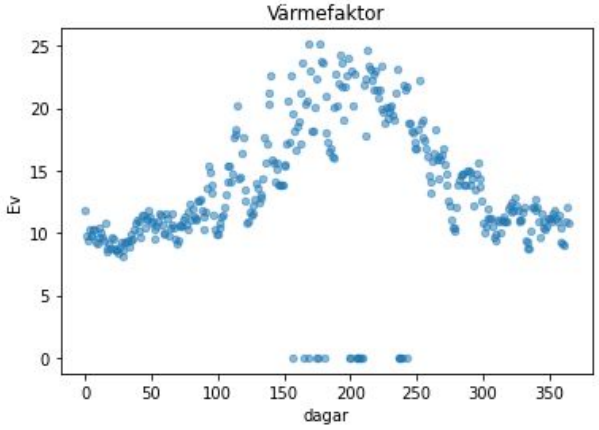


Långvågig strålning
(Jorden som system)





Värmefaktor
(Ordning och oordning i fysikaliska system)



Freonmängd
(Jorden som system)

Skriva text

```
print("Hello world")

# detta är en kommentar
```

Variabler

Likt matematiken

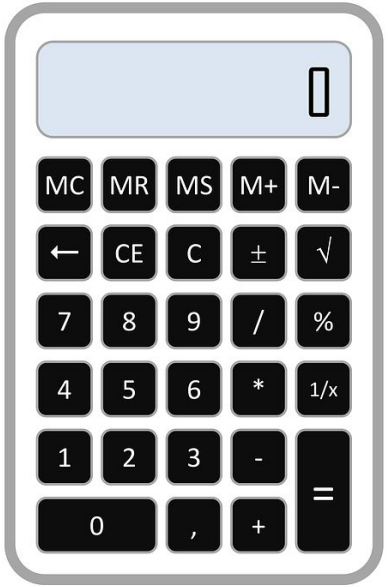
```
x = 5
y = "hej"
```

Sparar ett värde som kan användas flera gånger. Att få in en vana av att använda namn som betyder något kommer göra livet lättare i framtiden.

Matematiska operationer

Använda programmering som miniräknare

- Addition: +
- Subtraktion: -
- Multiplikation: *
- Division: /
- Upphöjt med: **
- Modulo: %



Uppgifter

“Instruktioner och variabler”

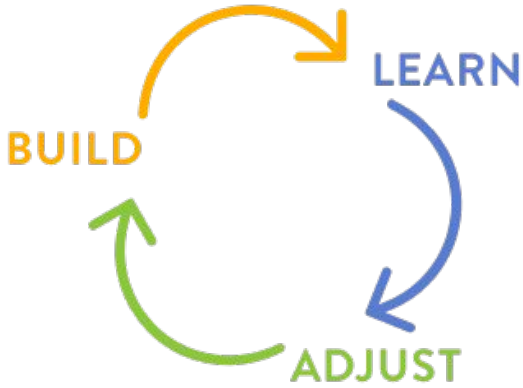
python.globalasystem.se

Andra timmen:

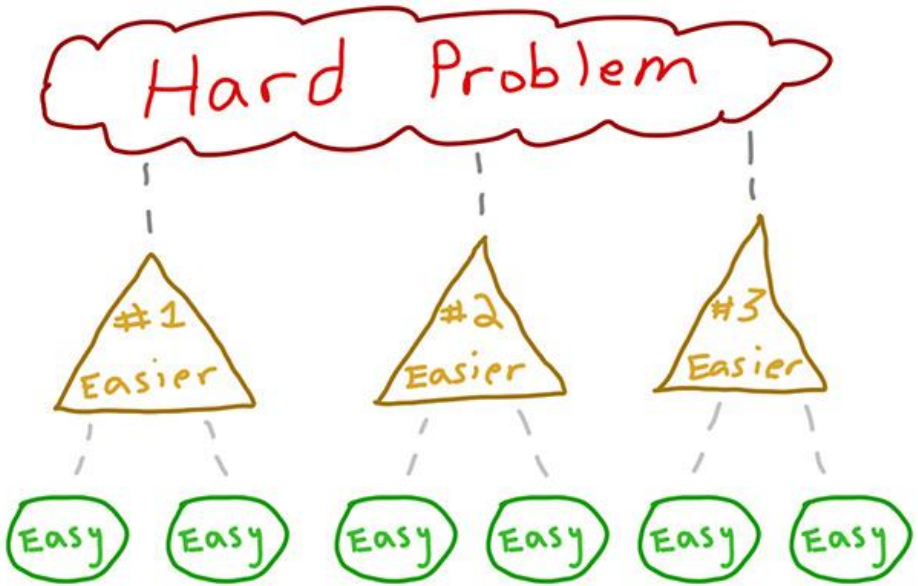
Arbetssätt, Datatyper & Listor

Jobba iterativt

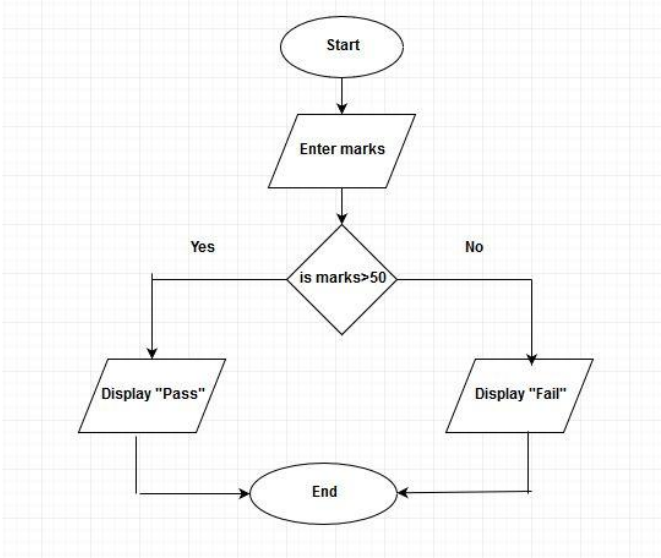
Fråga andra



Bryt ner problemet i mindre delar



Tankearbete innan man börjar skriva.



1. Ta emot information.
2. Värdera informationen.
3. Ge output beroende på fall.

Vart finns hjälp?

Datatyper

Int	positiva och negativa heltal
Float	positiva och negativa decimaltal
String	text
Boolean	sant eller falskt

Listor

```
x = [5, 7, 9]
```

x.append()	lägg till i listan
x.remove()	ta bort ett värde från listan
x.pop()	ta bort en plats i listan
len(x)	returnerar längden på listan
sum(x)	returnerar summan av alla tal i listan

Uppgifter

“Datatyper och listor”

python.globalasystem.se

Pythonintroduktion 2023

```
data = np.loadtxt('population.csv', dtype={'names': ('year', 'totpopulation'), 'formats': (np.int, np.float)}, delimiter=',', s


years = data['year']
pop = data['totpopulation']

#medelvärde procentuell ökning 1950-2019
pop_deriv = np.mean(np.diff(pop[-70:]) / pop[-1]) + 1

#medelvärde procentuell ökning 2010-2019
pop_deriv2 = np.mean(np.diff(pop[-10:]) / pop[-1]) + 1


#exponetiell regression (minsta kvadrat)
pop_reg = optimize.curve_fit(lambda


#år fram till 2050
futureYears = np.arange(2019,2051)

future_popC = np.empty(32)
future_popB = np.empty(32)
future_popA = pop_reg[0][0] * pop_reg[0][1] ** futureYears

for i in range(len(future_popC)):

    pop_i = pop[-1] * (pop_deriv ** (i))
    future_popC[i] = pop_i

    pop_i2 = pop[-1] * (pop_deriv2 ** (i))
    future_popB[i] = pop_i2

fig, ax = plt.subplots(1,1)
```

Villkor

If *this* then *that*

Villkor

Det finns flera logiska uttryck att använda i if-satser:

lika med	==
större än	>
mindre än	<
större än eller lika med	>=
mindre än eller lika med	<=
inte lika med	!=

and, or, not

Villkor

If-satser skriver vi så här:

```
x = 4

if x > 5:
    print('x är större än 5')
else:
    print('x är inte större än 5')
```

Villkor

If-satser skriver vi så här:

```
x = 4

if x > 5:
    print('x är större än 5')
elif x == 5:
    print('x är lika med 5')
else:
    print('x är mindre än 5')
```

Loopar

While *this* is true, do *that*

Uppgifter

“Villkor och loopar”

python.globalasystem.se

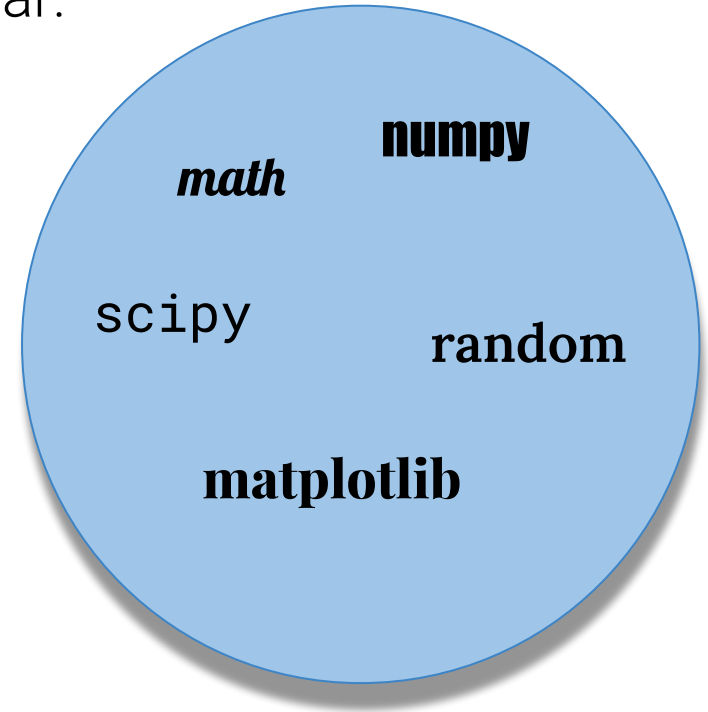
Andra halvtimmen:

1. Bibliotek
2. Tips och tricks!

Bibliotek

Används för mer avancerade beräkningar:

- Trigonometri
- Plotta grafer
- Naturliga logaritmen



Tips och tricks!

- ★ Skriv mycket kommentarer
- ★ Använd bra variabelnamn
- ★ Skriv tydlig kod
- ★ Gör mappar så att datorn hittar dina filer

Uppgifter

“Moduler och trigonometri”

python.globalasystem.se