

# Introduktion till programmering

DA354B 7.5HP HT24 LP2



# Anton Tibblin

Kursansvarig för kursen: DA354A – Introduktion till programmering

[anton.tibblin@mau.se](mailto:anton.tibblin@mau.se)    070-533 74 76

# Vem är jag?

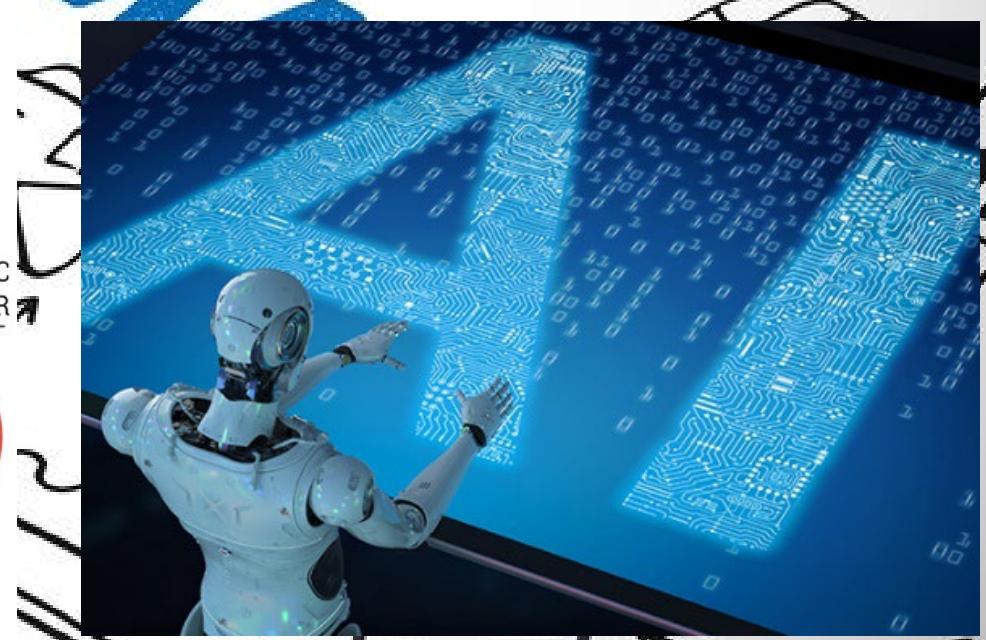
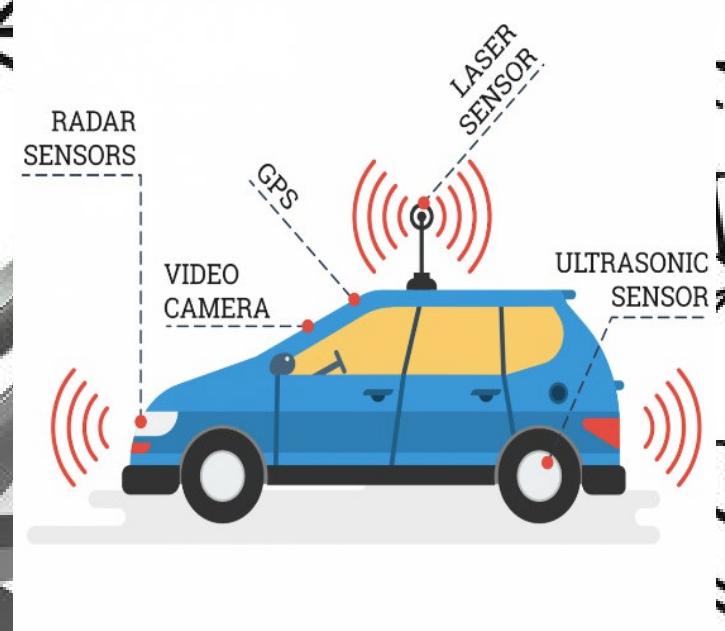
- Anton Tibblin
- Universitetsadjunk på Malmö universitet
- Programledare för IA-programmet
- Varit involverad i IA-programmet sedan 2012







<https://www.menti.com/albs7pg41m6w>



*Digitalisering* avser ursprungligen och i tekniska sammanhang omvandling från analog till digital representation av information med nollor och ettor, exempelvis genom sampling av ljud och mätsignaler eller genom skanning av bilder.

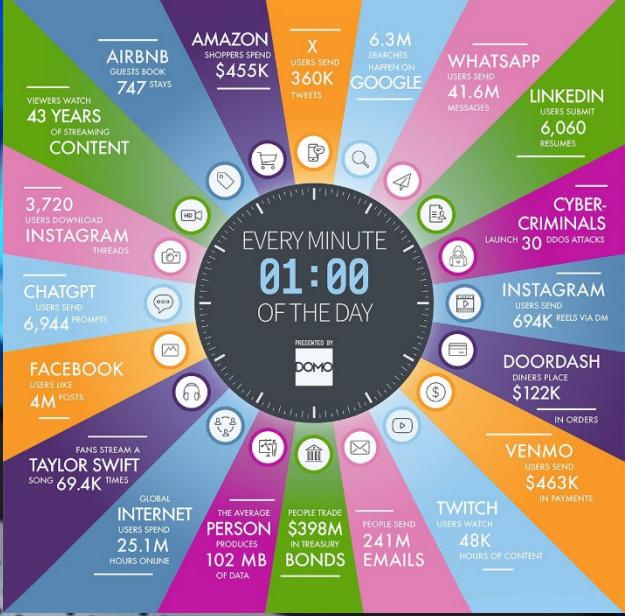


# Ett informationssamhälle

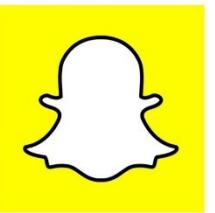
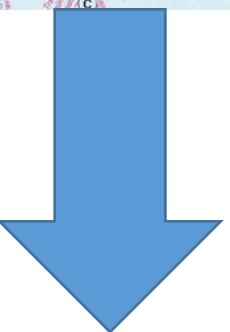
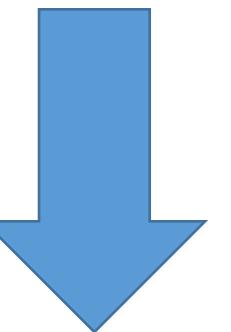
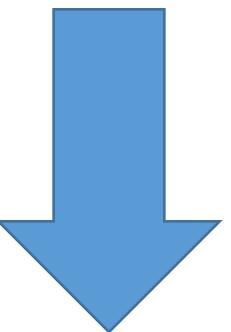












**Old School.**



**Highly  
Efficient.**

**VS.**





**NETFLIX**



©2006 All Rights Reserved

NETFLIX ORIGINAL

# OZARK

Spela upp

+ Min lista

**Se säsong 2 nu**

När det krisar måste familjen hålla ihop. Myskväll med familjen, det är nu att proppa väggarna fulla med knarkpengar.

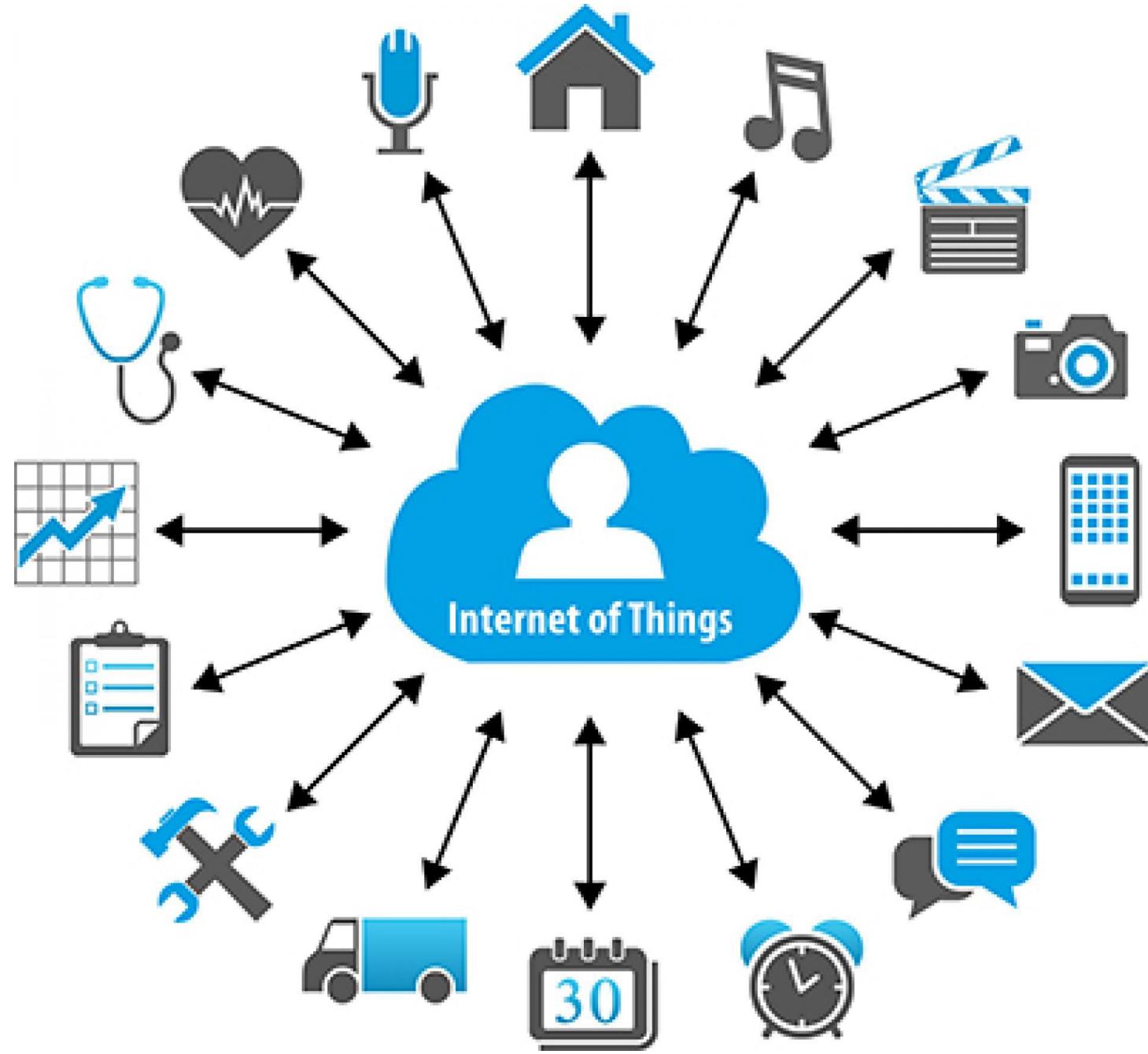
**ORIGINALINNEHÅLL FRÅN NETFLIX**

# Exempel: Skånetrafiken

- Här lägger jag specifikt fokus på biljetter som företeelse
  - Pappersbiljetten
  - Plastkortsbiljetten
  - Appbiljetten
  - Digitalt beroende verksamhet, värde kan svårlijgen skapas utan IT.



Foto: Carljohan Orre











# HOW DOES IT WORK





# SVENSKA DAGBLADET

[Start](#)[Näringsliv](#)[Kultur](#)[Ledare](#)[☰ Meny](#)[Debatt](#)

## Programmering är framtidens språk

Det är hög tid att fler unga får bekanta sig med programmering tidigt i livet. Det här är inget nytt. Flera länder har redan insett det och det är dags att även Sverige gör det, skriver Christine Johnson, IT-Gymnasiet.

BRÄNNPUNKT | IT I SKOLAN

”

I dag finns språket kod i nästan all teknik vi äger och använder oss  
av.

Christine Johnson

**Var med och  
påverka!**

Uttryck dina åsikter  
om reklam, media och  
varumärken.

Anmäl dig här

**För att Sverige ska stärka** sin konkurrenskraft och utvecklas till den kunskapsnation vi måste bli så behöver vi alla lära oss ett nytt språk: programkod (kod). Likaså viktig som engelskan varit de senaste 75 åren, kommer kod att bli för oss om bara några år.

**Debatt**

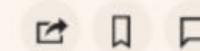
Det här är en argumenterande text med syfte att påverka.  
Åsikterna som uttrycks är

Ekonomistudion

# Brist på programmerare – saknas en miljon i EU

**SvD Näringsliv**

Publicerad 2018-04-23





DEBATT



## Så skickar du in dina debattartiklar!

Mejla inläggen till debatt-redaktionen

MEST LÄST PÅ DEBATT I DAG



Sätt farliga elever i särskilda skolor



Fridolins köttnationalism hotar miljön och klimatet

# Kod är makt och vi är framtidens analfabeter

Publicerad 22 okt 2014 kl 06.10

f Rekommendera

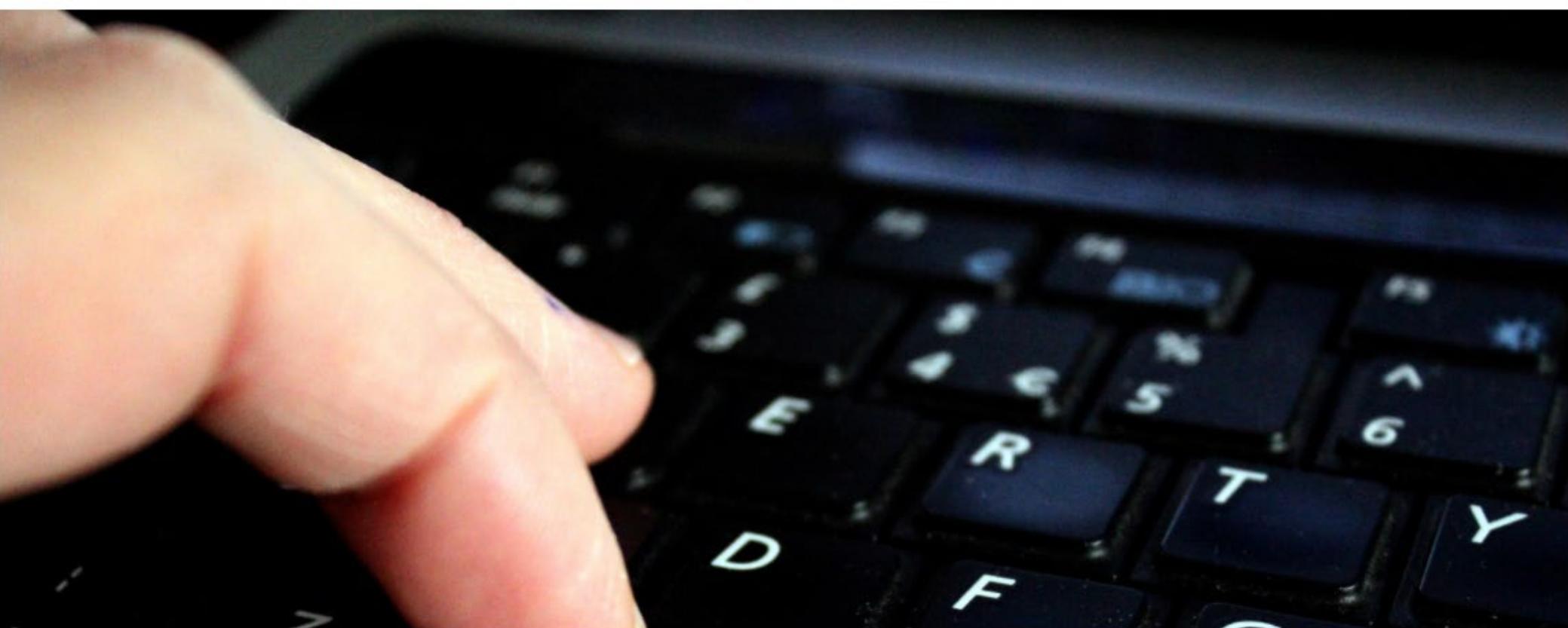
502

t Tweeta

g+ Dela

m Mejla

Den stereotypa 50-åriga universitetsutbildade mannen har fått konkurrens i



# Programmerare

[Start](#) / [Yrken](#) / [Data & IT](#) / [Programmerare](#)

År 2020 kommer Europa att sakna 1.000.000 programmerare (enligt prognos från EU-kommissionen).

Programmerare kan arbeta med allt ifrån dataspel till avancerade styrsystem för tillverkning i industrin. Vanligen arbetar du utifrån riktlinjer eller instruktioner som systemutvecklaren dragit upp eller diskuterar själv förutsättningarna med beställarna av programmet. Du måste vara beredd att lära dig nya språk när det krävs.

**Genomsnittlig lön\*** (Källa: SCB)

\* Angiven lön kan variera kraftigt beroende på efterfrågan, utbildningsnivå, yrkesfarenhet etc.

**42 800 kr**

↑ Lätt att få jobb!





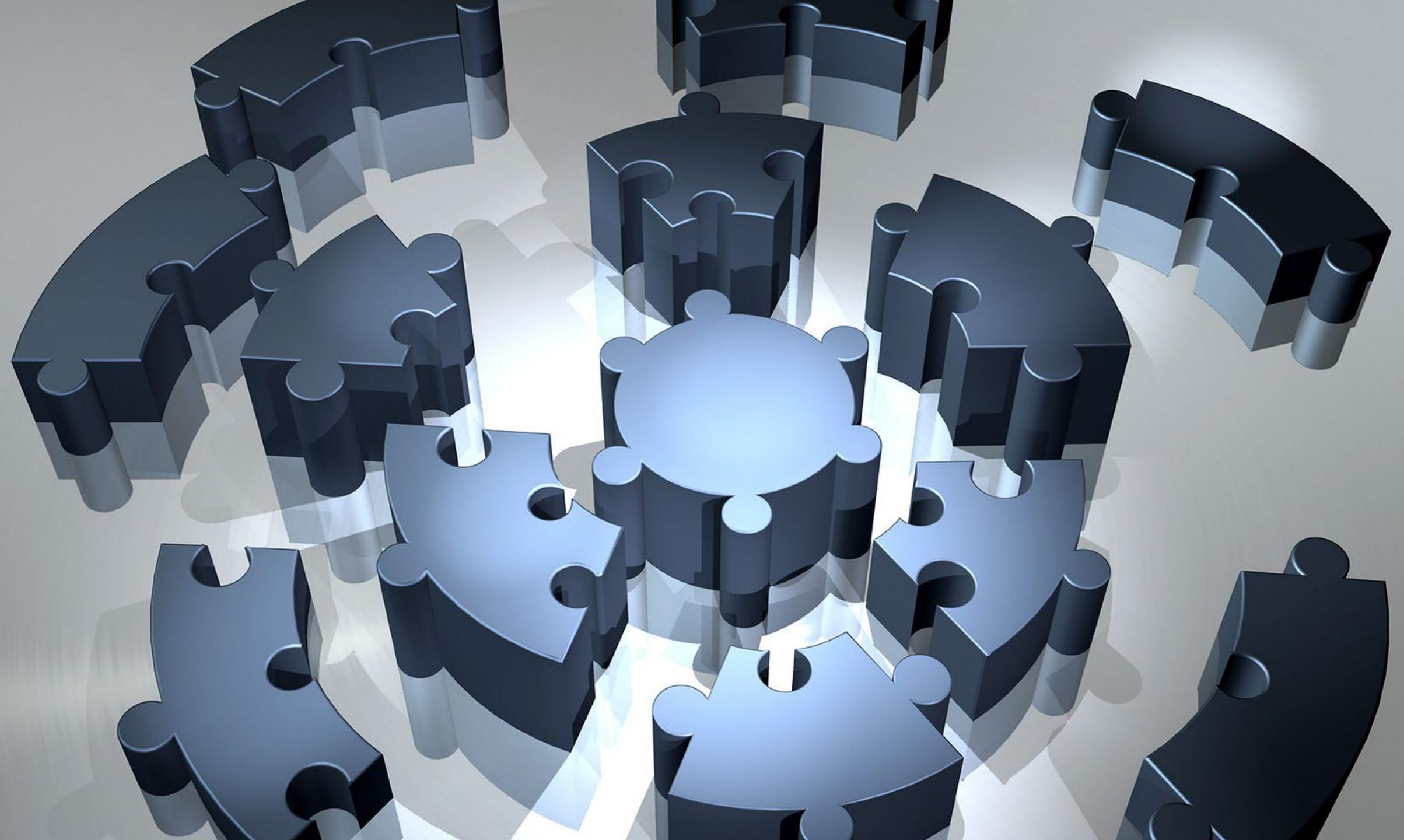
TALK  
<CODE>  
TO ME



TALK  
<CODE>  
TO ME



Vad går kursen ut på?





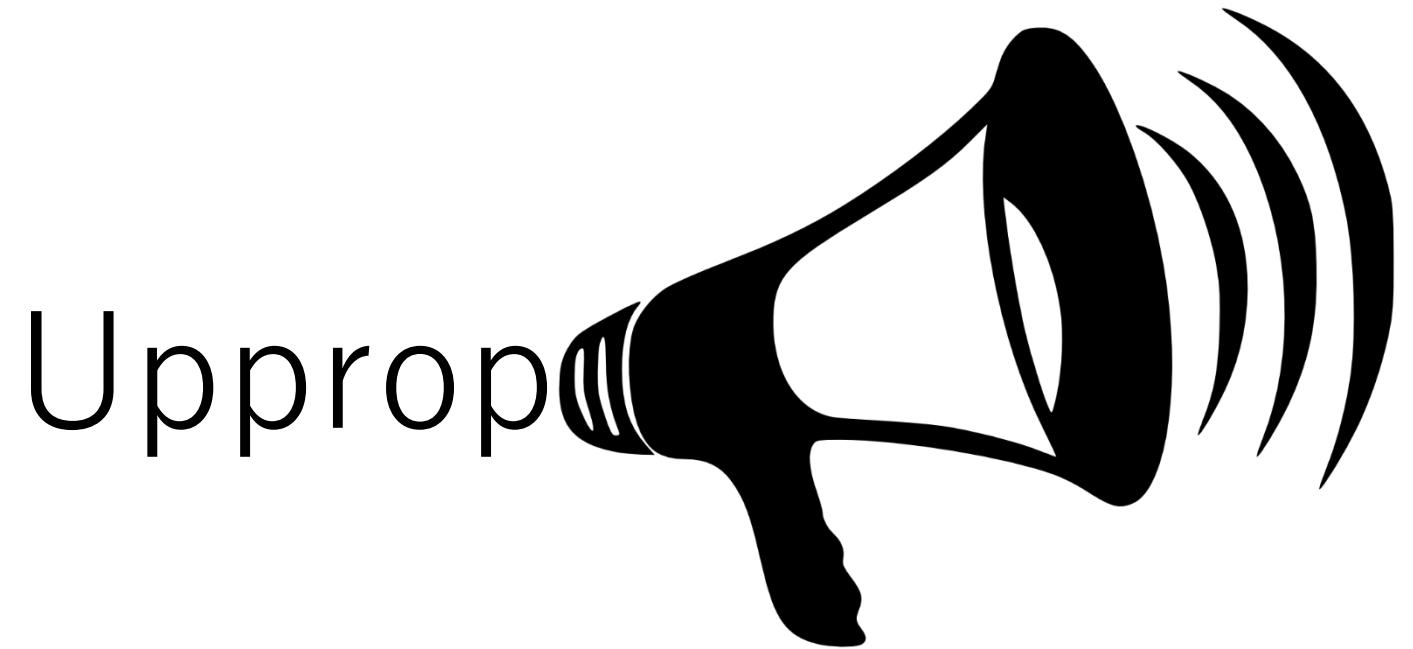
A close-up photograph of a person's face, focusing on their eyes and forehead. They are wearing dark-rimmed glasses. The person is looking directly at a computer screen whose content is visible through the lenses. The screen displays a large amount of code, likely JavaScript or similar, with various functions, variables, and syntax highlighted in different colors (e.g., blue for functions, red for strings). The code appears to be related to a tab or modal interface, given the presence of terms like "tab", "fade", and "active". The overall scene conveys a sense of someone deeply focused on coding or software development.

```
st a"),f=a.Event("hide.bs.tab  
faultPrevented()){var h=a(d);  
trigger({type:"shown.bs.tab",r  
u>.active").removeClass("ac  
ria-expanded",!0),h?(b[0].off  
().find('[data-toggle="tab"]  
de")||!d.find(">.fade").l  
;var d=a.fn.tab;a.fn.tab=b  
"show");a(document).on("br  
use strict";function b(b  
=typeof b&&e[b]())}var  
",a.proxy(this.checkPo  
null,this.pinnedOffset  
tState=function(a,b,c,  
"bottom"==this.affixe  
!-=c&&e<=c?"top":null  
.RESET).addClass("af  
WithEventLoop=funct  
ent.height(),d=this  
peof e&&(e=d.top(t  
ent.css("top",""),
```

# Dagens agenda

- Upprop
- Lärare & assistenter
- Kursens innehåll
- Kursplan
- Kursens planering
- Examinerande moment
- Kurslitteratur
- En historik programmeringsresa – Med Johan Holmberg





Upprop

# Lärare i kursen



- Anton Tibblin (kursansvarig)
  - [anton.tibblin@mau.se](mailto:anton.tibblin@mau.se)
  - 040-66 57672

- Arvid Berndtsson
  - Lärarassistent

- Teodor Esaiasson
  - Lärarassistent

# Kursens syfte

*Kursen syftar till att studenten utvecklar kunskap och färdigheter inom grundläggande programmering.*

*Under kursens gång inhämtas erfarenheter av problemanalys, problemlösning, implementering och dokumentation av programvara.*

# Kursens innehåll

- Kursen fokuserar på praktiska **programmeringsfördigheter**, förankrade i relevant teori. Utgångspunkten är studentens arbete med **specifika problem**. Kursen behandlar:
  - **problemlösningsstrategier**
  - användning av språkkonstruktioner från **strukturerad programmering**
  - **implementering** och utvärdering av egna program

# Lärandemål – Kunskap och förståelse

- kunna visa förståelse för **grundläggande begrepp** inom strukturerad **programmering**
- visa ett **algoritmiskt tankesätt** och förståelse för att använda informell logik vid utformandet av algoritmer
- kunna övergripande diskutera **skillnader och likheter mellan strukturerad och objektorienterad programering**

# Lärandemål – Färdighet och förmåga

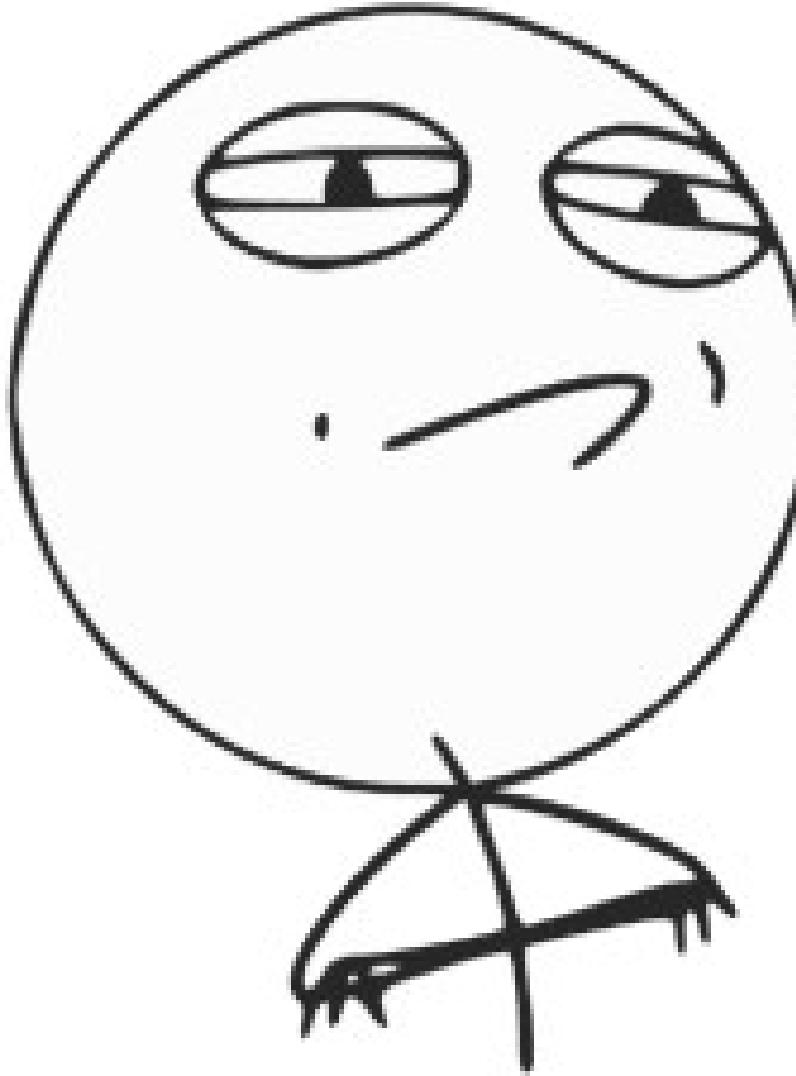
- kunna strukturera och **implementera enklare program** i ett programmeringsspråk
- självständigt från en **problembeskrivning** kunna **analysera** problem och **konstruera** ett fungerande program
- kunna motivera sin **programkod** med avseende på lämpliga **kontrollstrukturer, läsbarhet** samt **konventioner** i aktuellt programmeringsspråk
- kunna använda **tredjeparts- och standardbibliotek** vid programmering

# Lärandemål – Värderingsförmåga och förhållningssätt

- kunna **kritiskt resonera** kring alternativa lösningar för ett givet programmeringsproblem

# Problemlösning

# CHALLENGE ACCEPTED



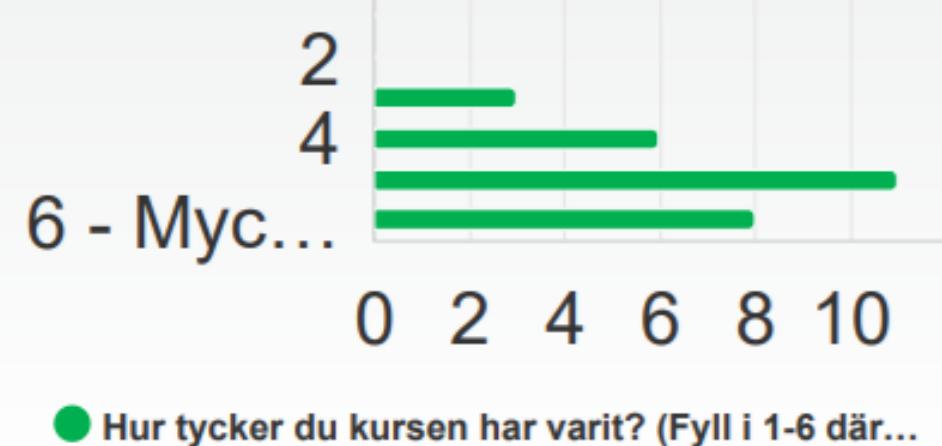
# Kursutvärdering

# HT2023

## 1 Hur tycker du kursen har varit? (Fyll i 1-6 där 1 är "Mycket dålig" och 6 är "Mycket bra")

Hur tycker du kursen har varit? (Fyll i 1-6 där 1 är "Mycket dålig" och 6 är "Mycket bra")

	Antal svar
1 - Mycket dålig	0 (0,0%)
2	0 (0,0%)
3	3 (10,7%)
4	6 (21,4%)
5	11 (39,3%)
6 - Mycket bra	8 (28,6%)
Summa	28 (100,0%)



## 4 Hur många timmar anser du att du har lagt ner på kursen per vecka?

Hur många timmar anser du att du har lagt ner på kursen per vecka?

	Antal svar
cirka 5 timmar eller färre	2 (7,1%)
cirka 10 timmar	3 (10,7%)
cirka 15 timmar	4 (14,3%)
cirka 20 timmar	12 (42,9%)
cirka 30 timmar	5 (17,9%)
cirka 30 timmar eller fler	2 (7,1%)
Summa	28 (100,0%)



● Hur många timmar anser du att du har lagt n...

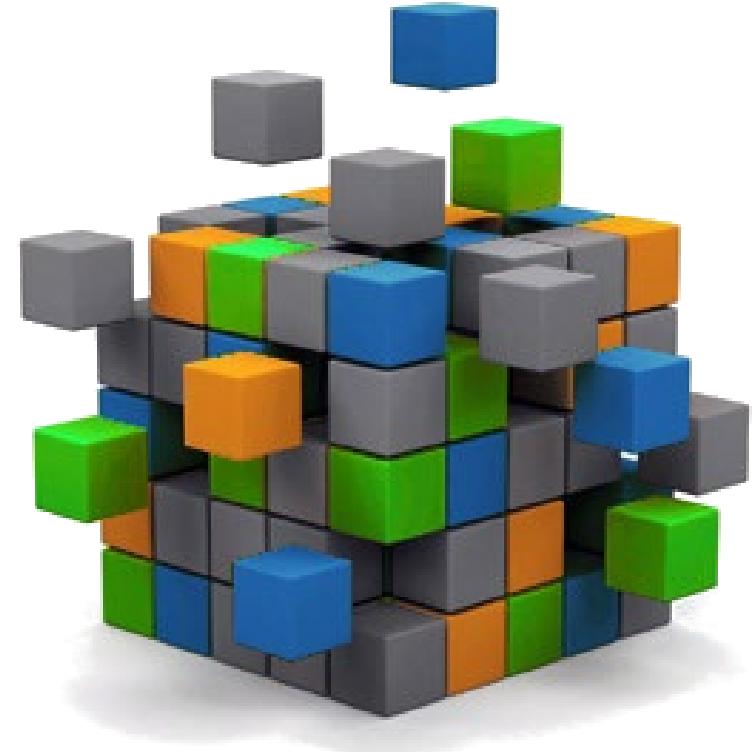
# Huvudpunkter från förra kursutvärderingen

- 1) Studenterna var generellt nöjda med kursen, läraren och kursens upplägg
- 2) Studenterna tyckte att examinationsformerna var bra
- 3) Studenterna tyckte att föreläsningarna var bra
- 4) Studenterna tyckte att laborationerna var bra
- 5) Studenterna önskade videoinspelningar (framförallt i början av kursen)
- 6) Önskar fler labbtillfällen och att de inte börjar 08.15 på morgonen
- 7) *Vi önskade att man gjort fler laborationer i par/grupper*
  - *Ni har själva möjlighet att lösa detta genom att skapa par/grupper när ni gör laborationerna.*
- 8) *Fokus på VS Code som editor (istället för IDLE)*
  - *På föreläsningarna kommer vi numera köra VS Code, samt att kursmaterialet uppdateras för detta.*



# Kursens moduler

1. Introduktion till Python
2. Funktioner
3. Iteration och selektion
4. Listor och lexikon
5. Fil- och felhantering
6. Webbapplikationer



**En modul per vecka, en inlämningsuppgift per vecka**

# Varje modul har..

- Föreläsningar
- Laborationer
- Inlämningsuppgifter
  - med undantag för första modulen

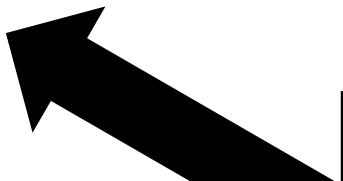


# Modulen börjar



47	18/11	10-12	F: Funktioner i Python	Anton	NI:A0406	
	19/11	13-15	F: Funktioner i Python (2)	Anton	OR:E439	2
	21/11	08-12	L: Funktioner i Python	Anton & Teo	OR:D326 & OR:E323	

# Modulen avslutas



# Examinerande moment

- Inlämningsuppgifter
- Deadlines: varje vecka från vecka 47.

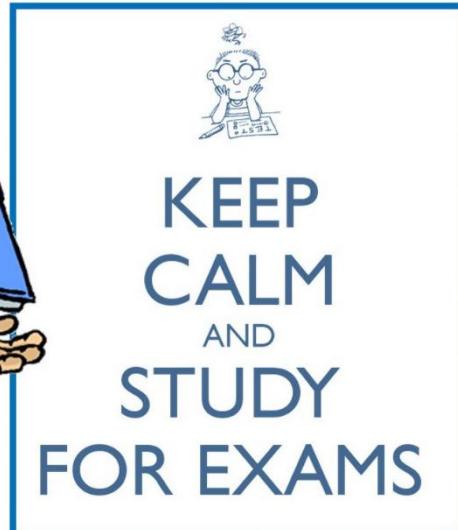
**1. 27/11**

**2. 4/12**

**3. 11/12 (Gruppuppgift)**

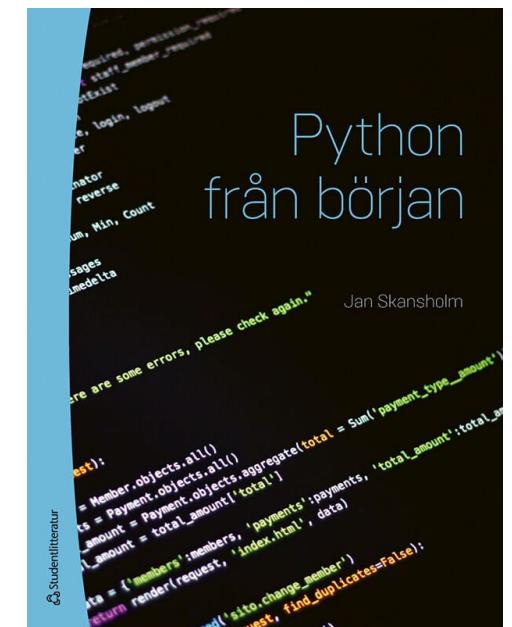
**4. 22/12**

**5. 19/1 < OBS. efter julen**



# Kurslitteratur

- Downey, Allen B. (2015) Think Python, O'Reilly, Beijing, ISBN-13: 9781491939369
  - <http://greenteapress.com/wp/think-python-2e/>
- <https://www.studentlitteratur.se/kurslitteratur/teknik/programmering/python-fran-borjan-40543-02>



<https://mau-webb.github.io/>  
Vår kurswebbplats

## Kursöversikt

[Planering](#)[Kursplan](#) ➔[Canvas](#) ➔[VS Code](#)

## 1. INTRODUKTION TILL PYTHON ^

[Översikt](#)[Kursintroduktion](#)[Föreläsning](#)[Ö.1: Python & VS Code](#)[Ö.2: Variabler & datatyper](#)[Ö.3: Debugging & kommentarer](#)[Ö.4: Case \(1\) - Valutakonverterare](#)[Ö.5: Case \(2\) - Budgetberäknare](#)[Ö.6: Case \(3\) - Valfritt](#)

# Introduktion till programmering (DA354A)

Denna webbplats innehåller material från en grundkurs i webbutveckling. De som läser [kursen](#) är den primära målgruppen, men mycket kan förhoppningsvis vara användbart även för andra. Allt material är publicerat under en [fri licens](#), vilket gör att det kan återanvändas av andra, så länge upphovsman anges.

Kurslitteratur i denna kurs är boken *Think Python* av Allen B. Downey. Ni kan den finnas gratis på nätet antingen som [webbsida](#) eller som [PDF](#). Föredrar du en fysisk kopia av boken kan du hitta en återförsäljare [här](#).

Om man föredrar en kursbok på svenska så kan vi rekommendera [Python från början](#), den kommer att fungera lika bra i denna kurser.

## Syfte

I den bakomliggande kursplanen beskrivs syfte och mål:

*Kursen syftar till att studenten utvecklar kunskap och färdigheter inom grundläggande programmering. Under kursens gång inhämtas erfarenheter av problemanalys, problemlösning, implementering och dokumentation av programvara.*

I denna kurs innebär det programmering i språket [Python](#). Kursen är uppdelad i 7 st moduler som progressivt bygger på varandra. Modulerna kommer att publiceras löpande under kursens gång, och en kort presenteras av modulerna hittar ni nedan.

## 1. Introduktion till Python

Modulen fokuserar på att komma igång med programmering i Python. Det innebär bl.a. att installera den mjukvara som behövs för att skriva och köra Python-kod, samt de mest grundläggande byggnestenarna i programmering som t.ex. *variabler, datatyper & utskrift* av data.

## 2. Funktioner

Modulen fokuserar på *vad* funktioner är för något, *varför* vi vill använda funktioner, samt *hur* vi skriver och använder funktioner i Python. Vi tittar även på *argument* och *parametrar* som gör att våra funktioner kan användas

Kursöversikt

[Planering](#)[Kursplan](#)[Canvas](#)[VS Code](#)**1. INTRODUKTION TILL PYTHON**[Översikt](#)[Kursintroduktion](#)[Föreläsning](#)[Ö.1: Python & VS Code](#)[Ö.2: Variabler & datatyper](#)[Ö.3: Debugging & kommentarer](#)[Ö.4: Case \(1\) - Valutakonverterare](#)[Ö.5: Case \(2\) - Budgetberäknare](#)[Ö.6: Case \(3\) - Valfritt](#)

# Planering

Varje modul är en vecka och introduceras varje måndag. Alltså börjar den första modulen måndagen den 11/11, den andra modulen den 18/11, den tredje modulen den 25/11, osv.

Inlämningsuppgifter som regel (undantag för inlämningsuppgift 5) har deadline onsdagen veckan efter modulen påbörjades, deadlines i kursen är:

- Modul 2: Inlämningsuppgift 1 har deadline onsdagen den 27/11
- Modul 3: Inlämningsuppgift 2 har deadline onsdagen den 4/12
- Modul 4: Inlämningsuppgift 3 har deadline onsdagen den 11/12 ( detta är en gruppuppgift)
- Modul 5: Inlämningsuppgift 4 har deadline söndagen den 22/12 (OBS. söndag)
- Modul 6: Inlämningsuppgift 5 har deadline söndagen den 19/1 (OBS. söndag)

Förkortning	Innebörd
F	Föreläsning
L	Laboration/övningar

Vecka	Datum	Tid	Moment	Lärare/Assistent	Plats	Modul
46	11/11	10-12	Kursintroduktion	Anton & Johan	NI:A0406	1
	12/11	13-15	F: Introduktion till programmering	Anton	NI:A0407	
	14/11	08-12	L: Introduktion till programmering	Anton & Teo	OR:D326 & OR:E323	
47	18/11	10-12	F: Funktioner i Python	Anton	NI:A0406	2
	19/11	13-15	F: Funktioner i Python (2)	Anton	OR:E439	
	21/11	08-12	L: Funktioner i Python	Anton & Teo	OR:D326 & OR:E323	
48	25/11	10-12	F: If-satser & Loopar (inspelad)	-	NI:A0407	

[Kursöversikt](#)[Planering](#)[Kursplan ↗](#)[Canvas ↗](#)[VS Code](#)**1. INTRODUKTION TILL PYTHON****Översikt**[Kursintroduktion](#)[Föreläsning](#)[Ö.1: Python & VS Code](#)[Ö.2: Variabler & datatyper](#)[Ö.3: Debugging & kommentarer](#)[Ö.4: Case \(1\) - Valutakonverterare](#)[Ö.5: Case \(2\) - Budgetberäknare](#)[Ö.6: Case \(3\) - Valfritt](#)

# Modul 1 - Introduktion till Python

## Översikt av modul 1

I den första modulen kommer vi att titta på några grundläggande koncept & byggstenar i programmering. Vi kommer framförallt att titta på:

- Variabler
- Datatyper
- Utskrifter
- Indata (från användaren)

Denna modul syftar till att introducera många av dessa koncept, i ett (för många) nytt programspråk, *Python*. Ö i tabell nedan, och menyn till vänster är förkortning för **övning**.

Innehåll	Kommentar
<a href="#">Kursintroduktion</a>	Kursens innehåll, struktur & upplägg
<a href="#">Föreläsning</a>	Introduktion till programmering
<a href="#">Ö.1: Python &amp; VS Code</a>	En introduktion till språket Python och miljön VS Code där vi skriver vår kod
<a href="#">Ö.2: Variabler &amp; datatyper</a>	En introduktion till grundläggande byggstenar i programmering
<a href="#">Ö.3: Debugging &amp; kommentarer</a>	En introduktion till att hitta (och åtgärda) fel i kod, samt kommentera kod
<a href="#">Ö.4: Case (1) - Valutakonverterare</a>	En något större uppgift att träna på
<a href="#">Ö.5: Case (2) - Budgetberäknare</a>	En något större uppgift att träna på
<a href="#">Ö.6: Case (3) - Extra övningar</a>	En något större uppgift att träna på
<a href="#">Inlämningsuppgift</a>	<i>Ingen uppgift i den första modulen</i>

## Rekommenderad läsning

- [Think Python - Chapter 1 The way of the program](#)
- [Think Python - Chapter 2 Variables, expressions and statements](#)

# Frågor?

<https://mau-webb.github.io/>  
Vår kurswebbplats

# Programdag IA

