

# **Introduktion till programmering**

**DA354A**

**7.5HP**

**HT19**

**LP2**

Go to **[www.menti.com](https://www.menti.com)** and use the code **16 69 87**



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



# Ett informationssamhälle



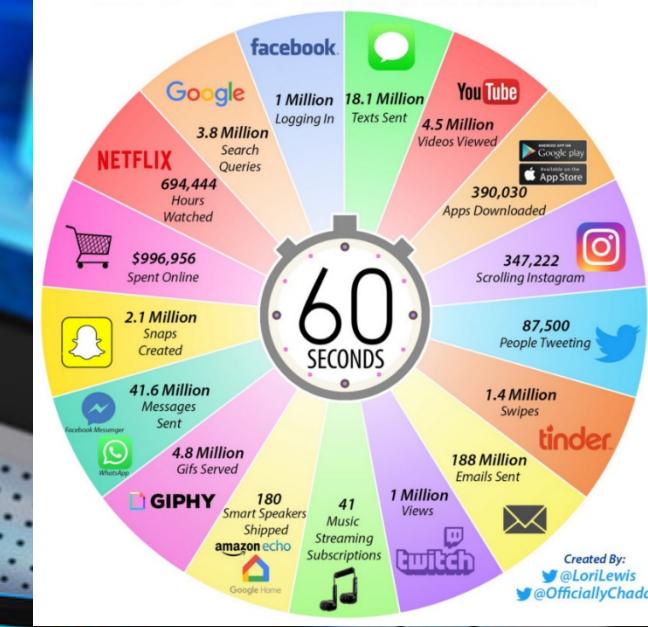


**En väg framåt i utvecklingen**

# 2019 *This Is What Happens In An Internet Minute*

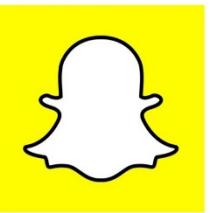
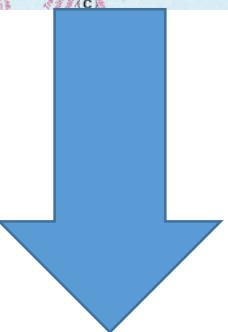
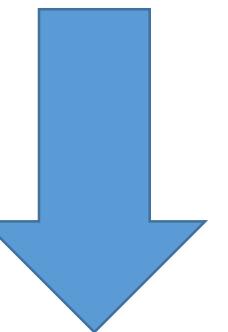
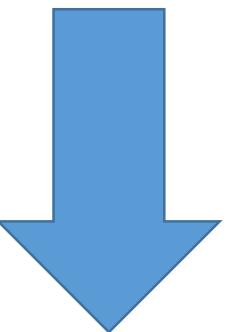


# 2019 This Is What Happens In An Internet Minute









**Old School.**



**Highly  
Efficient.**

**VS.**





**NETFLIX**



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.

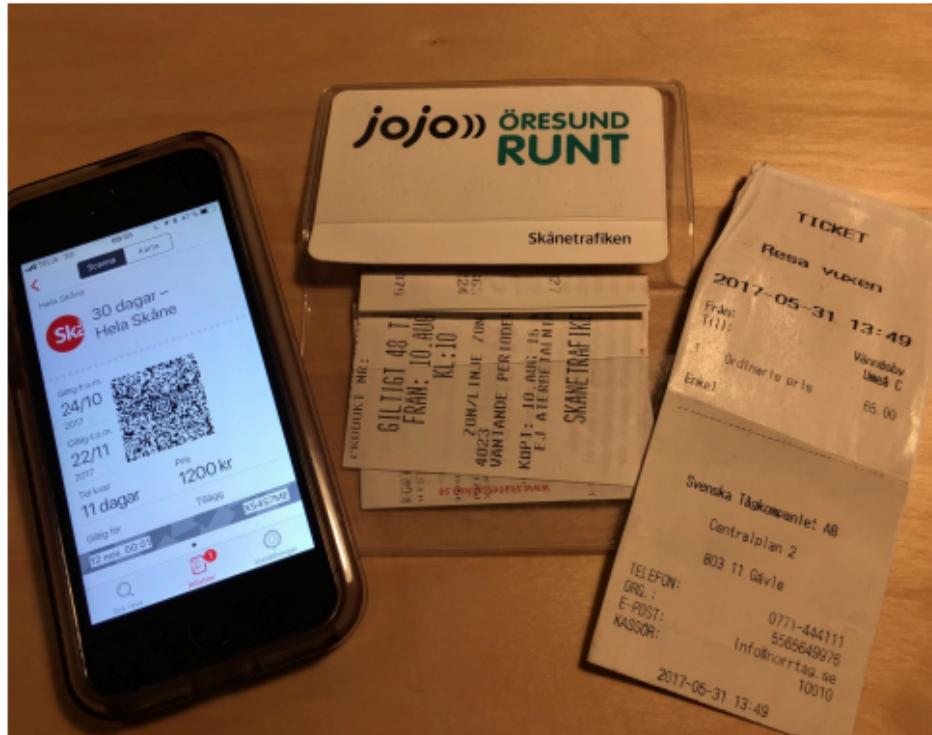
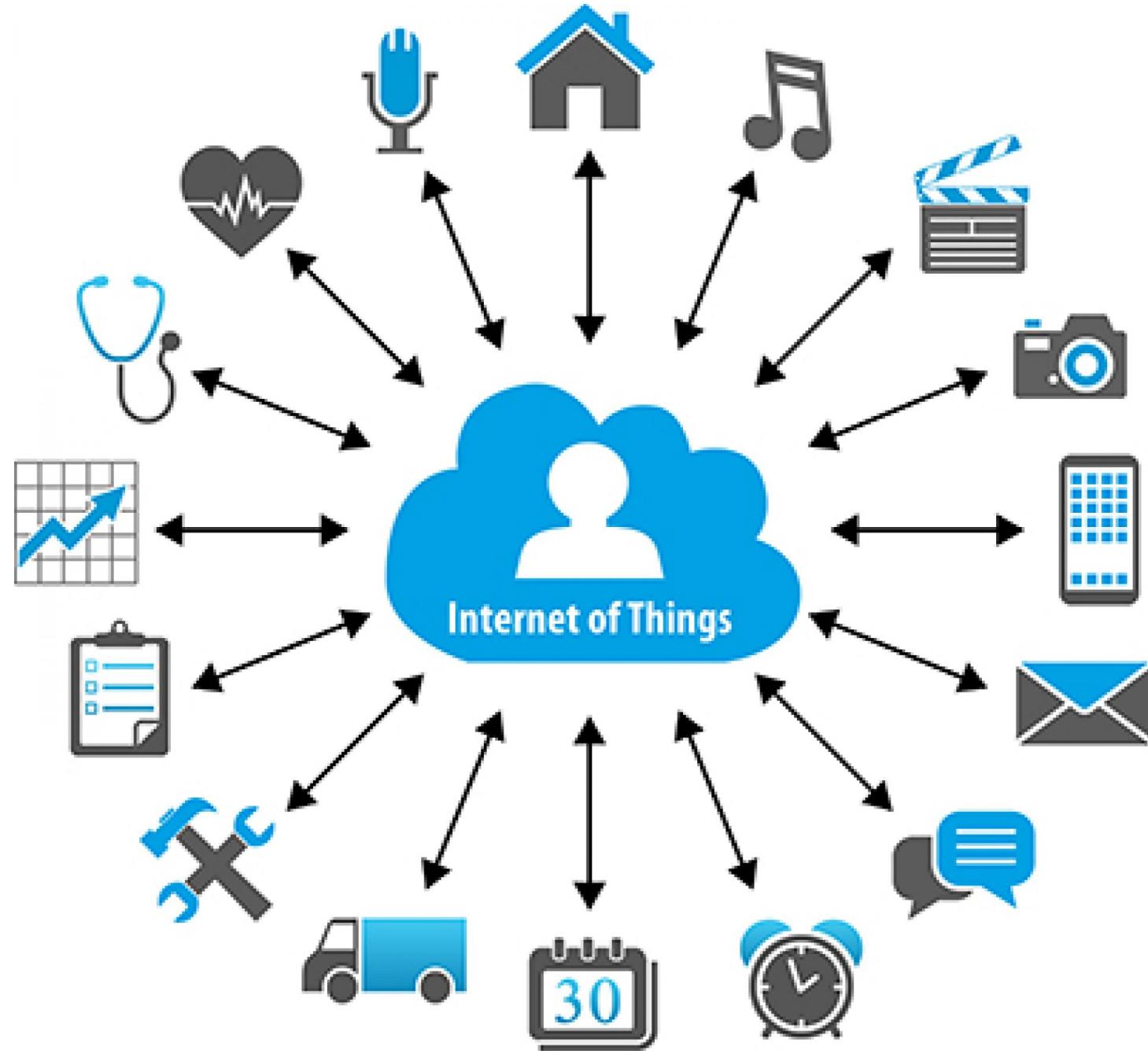


Foto: Carljohan Orre













**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

**Ekonomitudion**

# 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

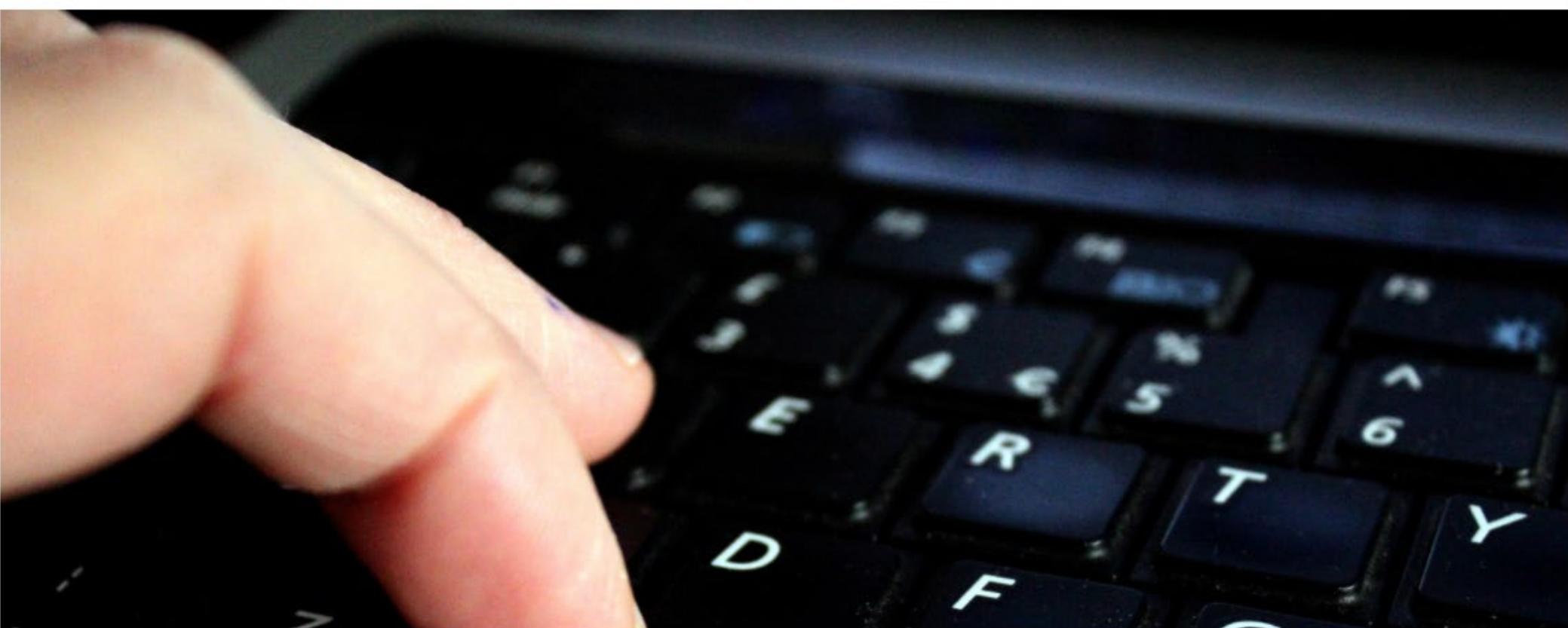
502

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.

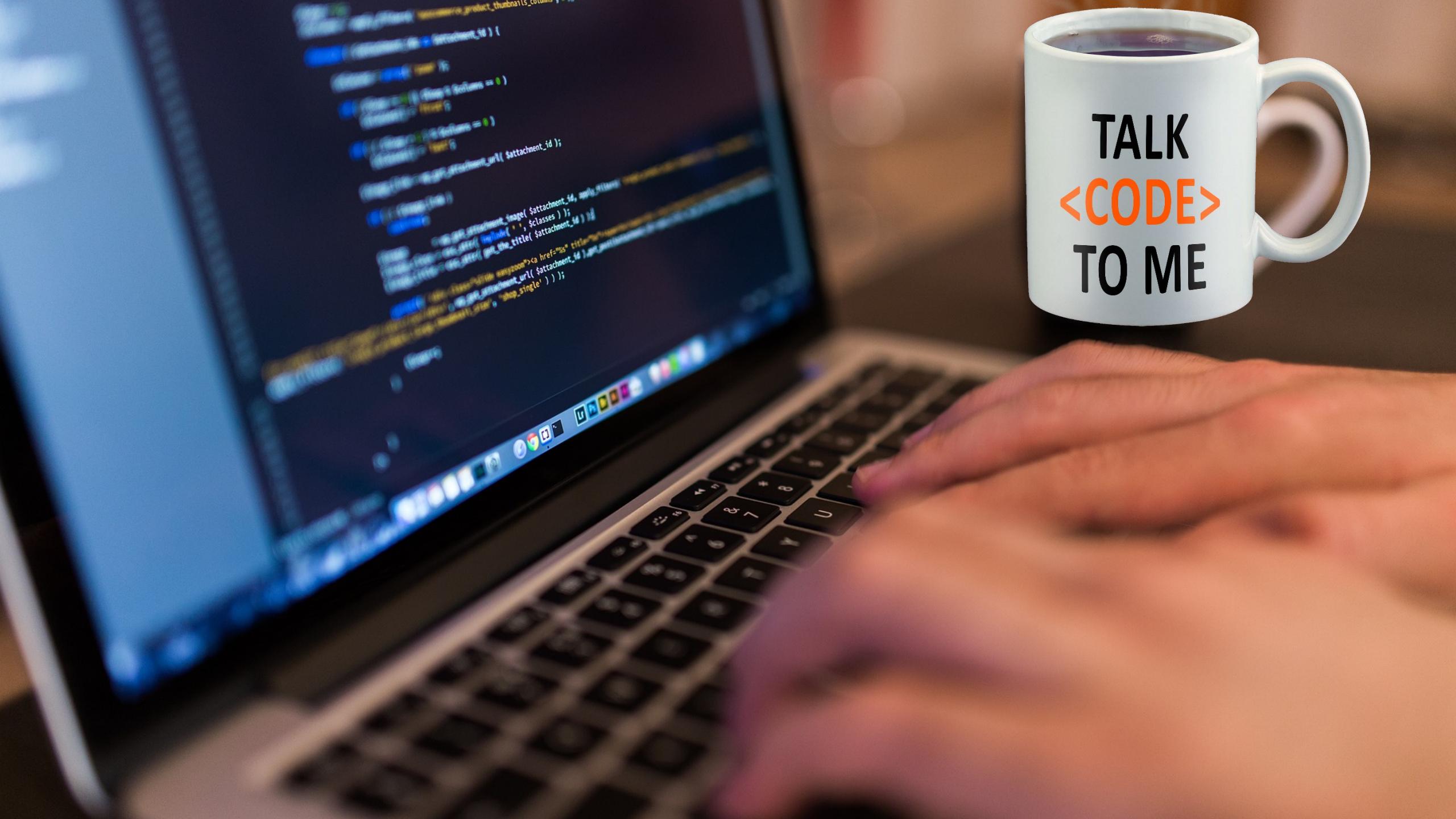
**42 800 kr**

↑ Lätt att få jobb!





TALK  
**<CODE>**  
TO ME

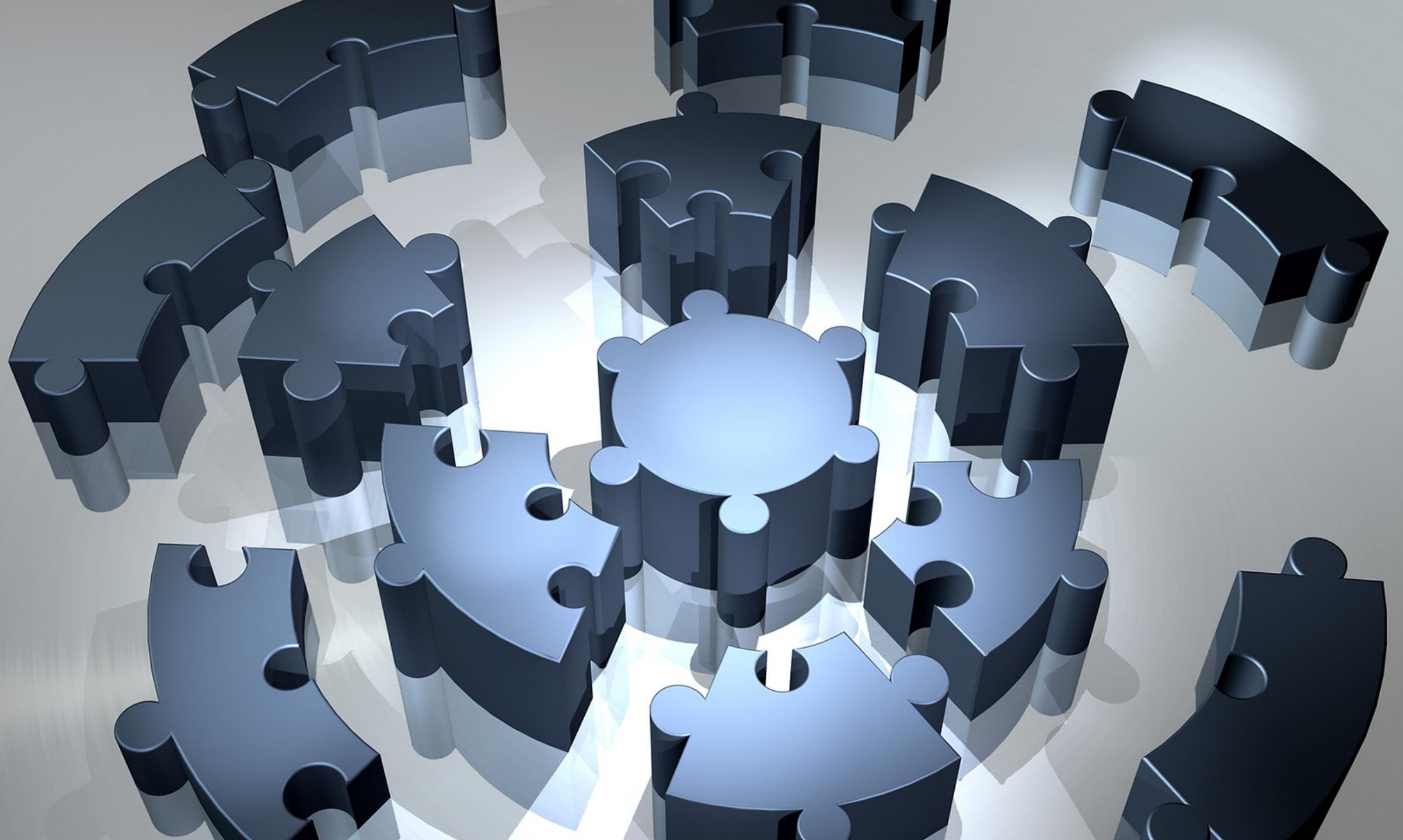


TALK  
<CODE>  
TO ME



Vad går kursen ut på?









# Dagens agenda

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



**Upprop**



# Vilka lärare är med i kursen?

- Lärare
  - Anton Tibblin
- Assisterter
  - Jesper Pamp
  - Zandra Nilsson



# 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

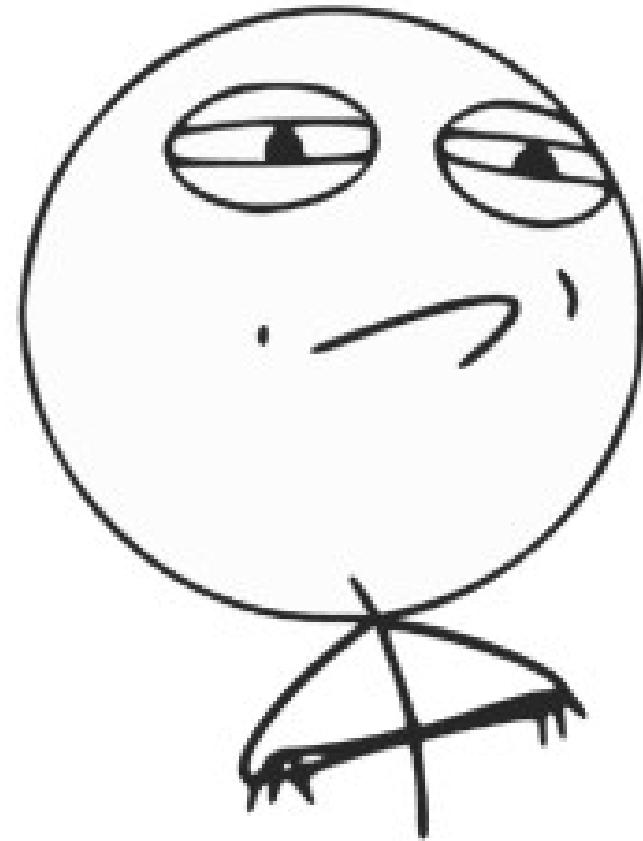
- kunna strukturera och **implementera enklare program** i ett programmeringsspråk
- självständigt från en **problembeskrivning** kunna **analysera** problem och **konstruera** ett fungerande program
- kunna motivera sin **programkod** med avseende på lämpliga **kontrollstrukturer, läsbarhet** samt **konventioner**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

# Lärandemål – Värderingsförmåga och förhållningssätt

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

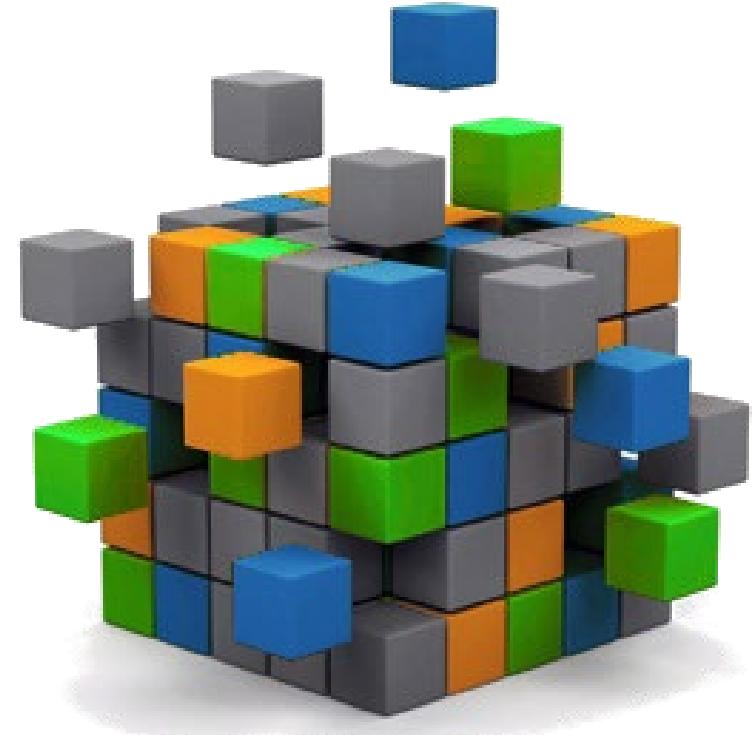
# Problemlösning

# CHALLENGE ACCEPTED



# 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



# Examinerande moment

- Inlämningsuppgifter
- Deadline: söndagar varje vecka från vecka 48. Deadlines blir alltså följande:
  1. 1/12
  2. 8/12
  3. 15/12
  4. 22/12
  5. 12/1



# Kurslitteratur

- Downey, Allen B. (2015) Think Python, O'Reilly, Beijing, ISBN-13: 9781491939369
- <http://greenteapress.com/wp/think-python-2e/>



**da354a.ia-mau.se**

Vår kurswebbplats

# Introduktion till programmering för IA

## Kursens planering

EN ÖVERSIKT AV KURSENS MOMENT

## Modul 1 - Introduktion till Python

EN INTRODUKTION TILL PYTHON OCH DESS UTVECKLINGSMILJÖ

Användning av allt material är fritt, om  
motsatsen inte anges. Formellt utrycks detta  
med licensen [Creative Commons Erkännande](#)  
CC BY

Detta projekt är publicerat som [öppen källkod](#)  
[på GitHub](#). Föreslå gärna ändringar och bidra  
med förbättringar genom att göra en [fork](#) av  
[denna projekt på GitHub](#)

- [Hem](#)
- [Kontakt](#)

# Kursens planering

För enkelhetens skull är alla viktiga datum och tider sammanställda här. För att undvika motsägande information är salar inte angivna - [se det officiella schemat](#).

Labbtillfällena är i grunden fria att disponera, men i tabellen nedan anges så långt man minst bör sträva efter att komma. (Det skadar aldrig att jobba senare delar!)

Vecka	Datum	Tid*	Vad
46	13/11	13-15	Kursintroduktion
	15/11	08-10, 10-12	Introduktion till Python + <b>laboration</b>
47	19/11	08-12	Laboration 1: <b>Introduktion till Python</b>
	20/11	13-17	Föreläsning + laboration - Funktioner
48	26/11	08-12	Laboration 2: Funktioner
	27/11	13-15	Föreläsning - If-satser och loopar
	29/11	09-10, 10-12	Inför + laboration 3: If-satser och loopar
	1/12		<b>Deadline: Inlämningsuppgift 1</b>
49	3/12	8-12	Laboration 3: If-satser och loopar
	4/12	13-15	Föreläsning - Listor och lexikon
	6/12	09-10, 10-12	Inför + laboration 4: Listor och lexikon
	8/12		<b>Deadline: Inlämningsuppgift 2</b>

## Modul 1 - Introduktion till Python

Övningar +

1. Bekanta dig med Python & IDLE
2. Skriva Python-kod i egna filer
3. Variabler & datatyper
4. Användarinput i Python
5. Ett valutakonverteringsprogram
6. Identifiera felet i koden
7. Extra, Slutövning

# Modul 1 - Introduktion till Python

## Översikt av modul 1

I modul 1 kommer vi att titta på några grundläggande koncept & byggstenar i programmering. Vi kommer att titta på hur man kan:

- lagra data, bland annat av typerna heltal och textsträng, i variabler,
- ta emot input från användaren, göra olika typer av beräkningar baserat på detta, sedan presentera resultat i form av output,
- styra programflöde med hjälp av selektion (if / else if / else) och iteration (for och while-loopar),
- organisera programdelar i fristående funktioner, av olika abstraktionsnivå,
- och mera!

Denna laboration syftar till att introducera många av dessa koncept, i ett (för många) nytt programspråk, *Python*.

## Instruktioner

Denna laboration är uppdelad i fyra delar,

### Del 1 - Introduktion till Python & IDLE

En inledning där du får bekanta dig Pythons interaktiva exekveringsfönster, samt med IDLE, programmeringsmiljön som följer med Python.

Frågor?