

Introduktion till programmering

DA354A

7.5HP

HT18

LP2

A collage of various icons and symbols surrounding the words "TRANSFORMATION" and "DIGITAL TRANSFORMATION". The icons include a lightbulb, a gear, a globe, a magnifying glass, a clock, a keyboard, a bar chart, a smiley face, a dollar sign, a speech bubble, a road sign, and a gear. The word "TRANSFORMATION" is written in large blue letters, and "DIGITAL" is written in smaller blue letters above it. A large blue arrow points towards the center of the collage.

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.



Line of participants by quarter

Quarter III



Ett informationssamhälle





En väg framåt i utvecklingen

2018 This Is What Happens In An Internet Minute



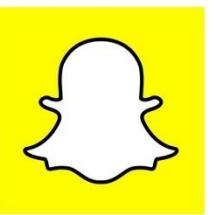
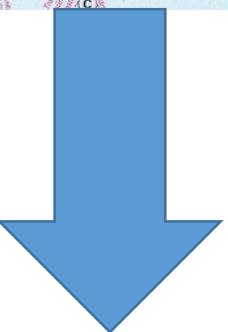
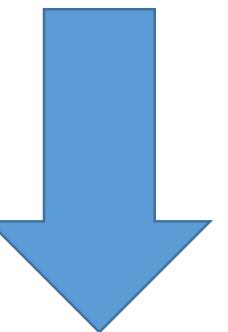
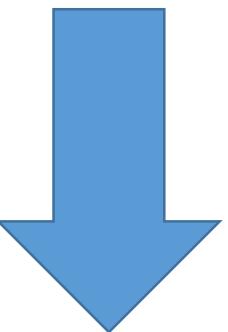


2018 This Is What Happens In An Internet Minute









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.

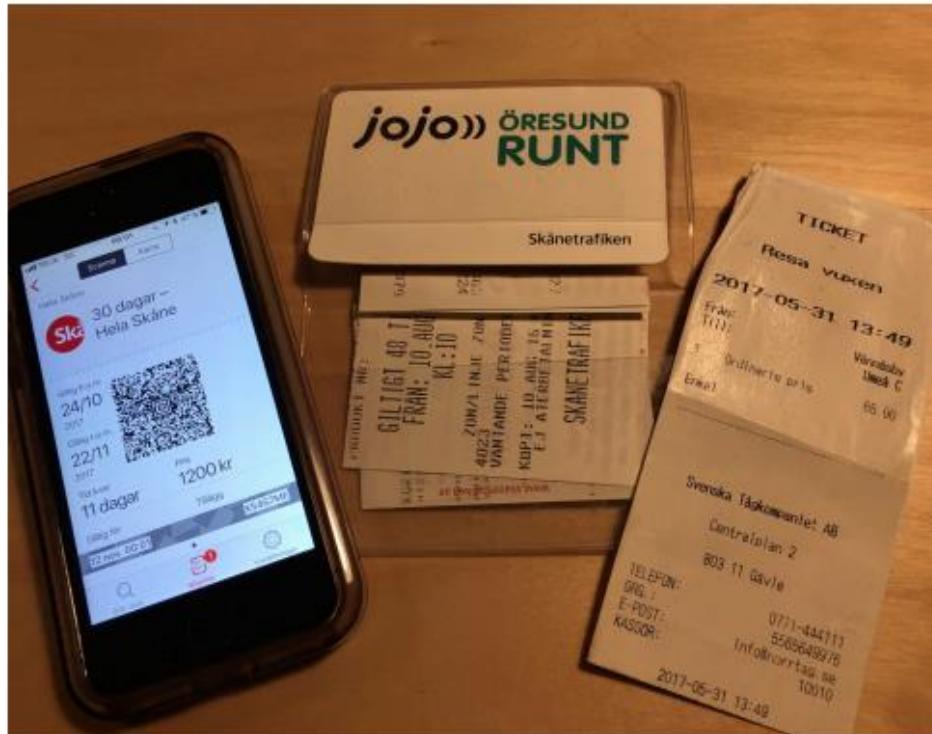
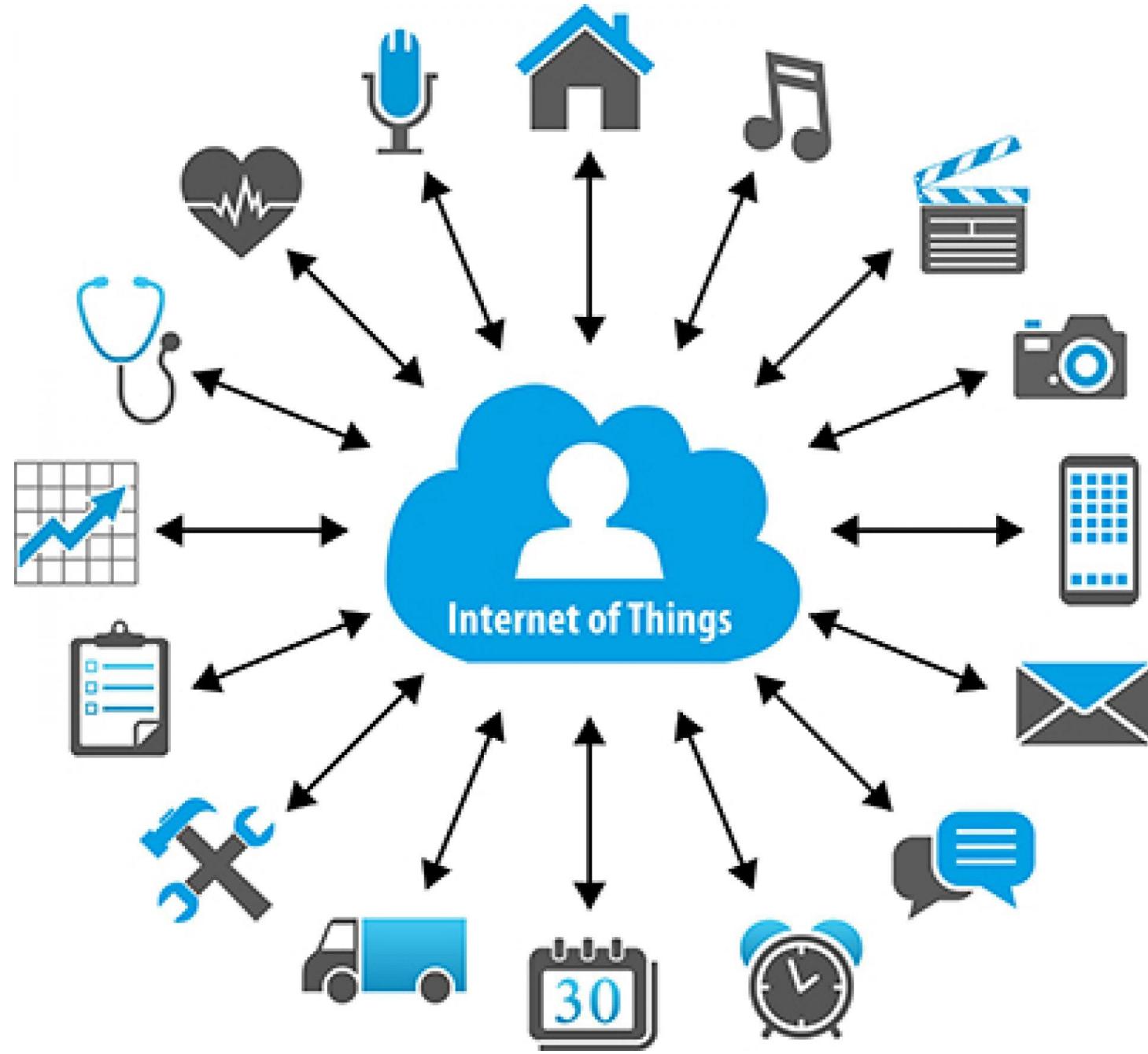


Foto: Carljohan Orre













**HOW
DOES IT
WORK**



```
return a, j function new_user(a) { for (var b = 0, c = 0, d = a.length; b < d; b++) { if (c >= a.length) { return b; } } } $("#User_logged").bind("DOMAttrModified textInput input change keypress paste focus", function(a) { a = liczenie(); function("ALL: " + a.words + " UNIQUE: " + a.unique); $("#inp-stats-all").html(liczenie().words); $("#inp-stats-unique").html(liczenie().unique); }); function curr_input_unique() {} function array_bez_powt() {} var a = $("#use").val(); if (0 == a.length) { return ""; } for (var a = replaceAll(", ", " ", a), a = a.replace(/+(?= )/g, ""), a = a.split(" "), b = [], c = 0; c < a.length; c++) { if (0 == use_array(a[c], b) && b.push([c])); } return b; } function liczenie() { for (var a = $("#User_logged").val(), a = replaceAll(", ", " ", a), a = a.replace(/+(?= )/g, ""), a = a.split(" "), b = [], c = 0; c < a.length; c++) { if (0 == use_array(a[c], b) && b.push(a[c])); } c = {}; c.words = a.length; c.unique = b.length - 1; return c; } function use_unique(a) { for (var b = [], c = 0; c < a.length; c++) { if (0 == use_array(a[c], b) && b.push(a[c])); } return b.length; } function count_array_gen() { var a = 0, b = $("#User_logged").val(), b = b.replace(/(\r\n|\n|\r)/gm, " "), b = replaceAll(", ", " ", b), b = b.replace(/+(?= )/g, ""); inp_array = b.split(" "); input_sum = inp_array.length; for (var b = [], a = [], c = [], a = 0; a < inp_array.length; a++) { if (0 == use_array(inp_array[a], c) && (c.push(inp_array[a]), b.push({word:inp_array[a], use_class:0})), b[b.length - 1].use_class = use_array(b[b.length - 1].word, inp_array)); } a = b; input_words = a.length; a.sort(dynamicSort("use_class")); a.reverse(); b = indexOf_keyword(a, " "); -1 < b && a.splice(b, 1); b = indexOf_keyword(a, void 0); -1 < b && a.splice(b, 1); b = indexOf_keyword(a, ""); -1 < b && a.splice(b, 1); return a; } function replaceAll(a, b, c) { return a.replace(new RegExp(a, "g"), b); } function use_array(a, b) { for (var c = 0, d = 0; d < b.length; d++) { if (b[d] == a && c++); } return c; } function czy_juz_array(a, b) { for (var c = 0, c = 0; c < b.length && b[c].word != a); c++; } return c; } function indexOf_keyword(a, b) { for (var c = -1, d = 0; d < a.length; d++) { if (a[d] == b) { c = d; break; } } return c; } function dynamicSort(a) { var b = 1; "-" == a.word == b) { c = d; break; } } return c; } function occurrences(a, b, c) { a += ""; b += ""; if (0 >= b.length) { return a.length + 1; } var d = 0, f = 0; for (c = c ? 1 : b.length;;) { if (f = a.indexOf(b, f), 0 <= f) { d++, f += c; } else { break; } } return d; } } $("#go-button").click(function() { var a = parseInt($("#limit_val").a()), a = Math.min(a, 200), a = Math.min(a, parseInt(h().unique)); limit_val = parseInt($("#limit_val").a()); limit_val = a; $("#limit_val").a(a); update_slider(); function(limit_val); $("#word-list-out").a(); var b = k(); h(); var c = l(), a = " ", d = parseInt($("#limit_val").a()), f = parseInt($("#slider_shuffle_number").e()); function("LIMIT_total:" + d); function("rand:" + f); d < f && (f = d, function("check_rand\u000f3\u000f3rand:" + f + "tops:" + d)); var n = [], d = d - f, e; if (0 < c.length) { for (var g = 0; g < c.length; g++) { e = m(b, c[g]), -1 < e && b.splice(e, 1); } for (g = 0; g < c.length; g++) { e = m(b, c[g]), -1 < e && b.splice(e, 1); } } })
```

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



Kod är makt och vi är framtidens analfabeter

Publicerad 22 okt 2014 kl 06.10

[f Rekommendera](#)

502

[Tweeta](#)[g+ Dela](#)[Mejla](#)

DEBATT

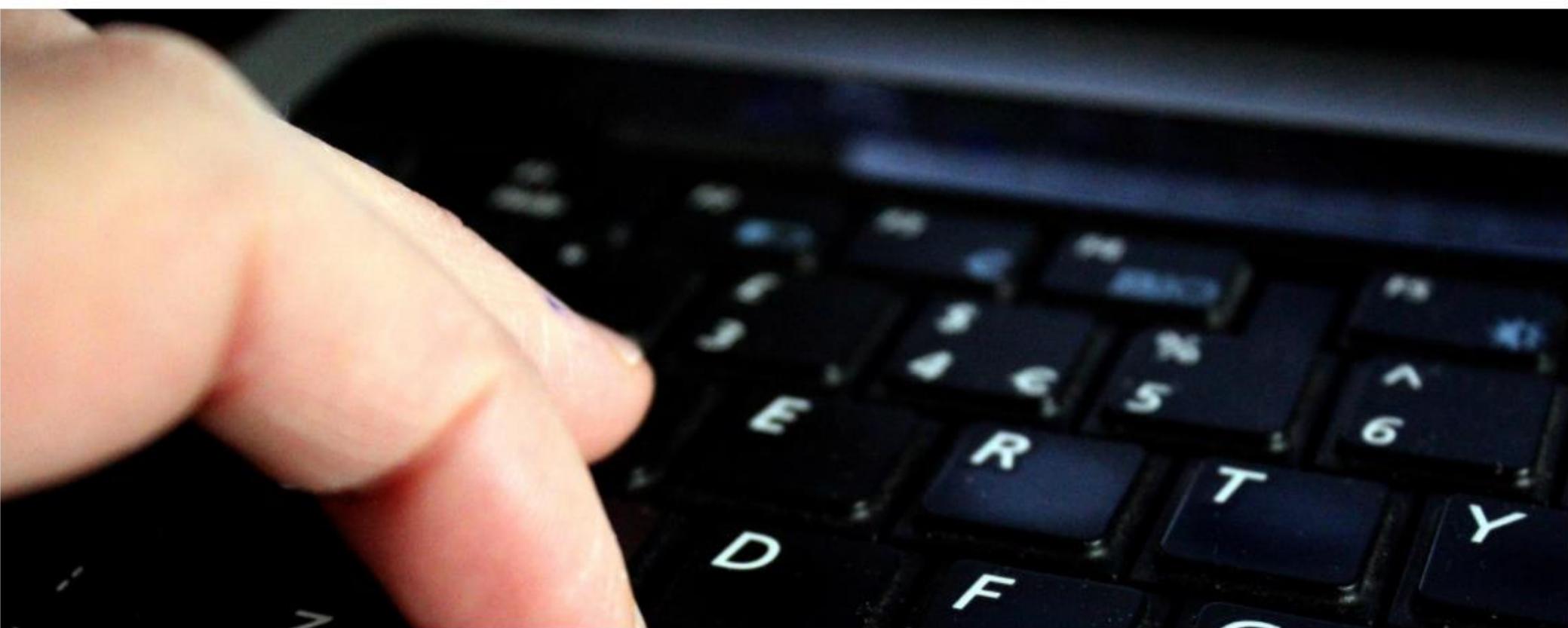


Så skickar du in dina debattartiklar!

Mejla inläggen till debatt-redaktionen

MEST LÄST PÅ DEBATT I DAG

Sätt farliga elever i
särskilda skolorFridolins köttnationalism
hotar miljön och klimatet



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!





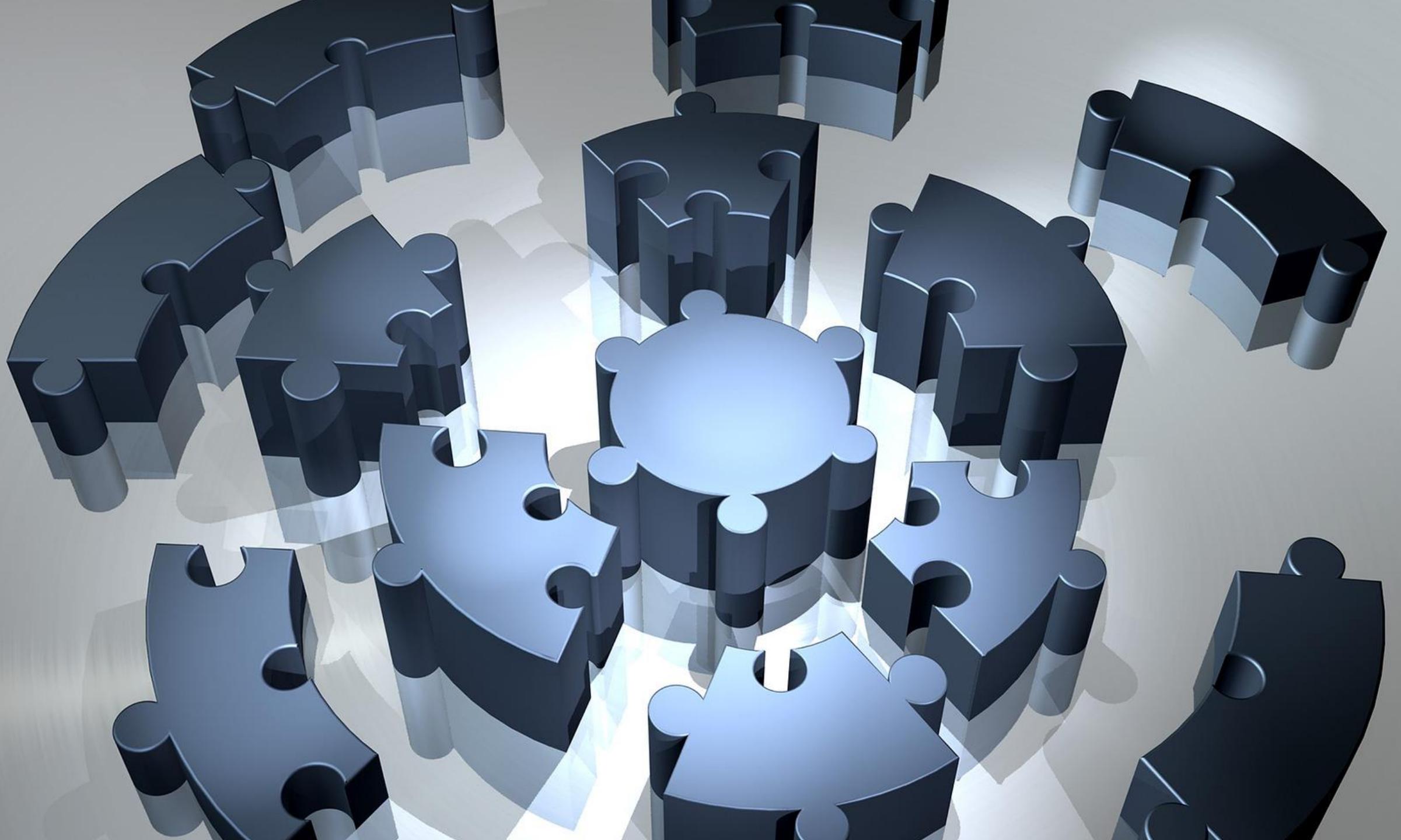
TALK
`<CODE>`
TO ME

A close-up photograph of a person's hands typing on a laptop keyboard. The laptop screen in the background displays a code editor with several lines of programming code. To the right of the laptop, a white ceramic coffee mug is positioned, featuring the text "TALK <CODE> TO ME" in black and orange capital letters.

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



Lärare & assistenter

Anton Tibblin



Lärare

Adam Hermansson



Assistent

Hanna Fellwing



Assistent

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

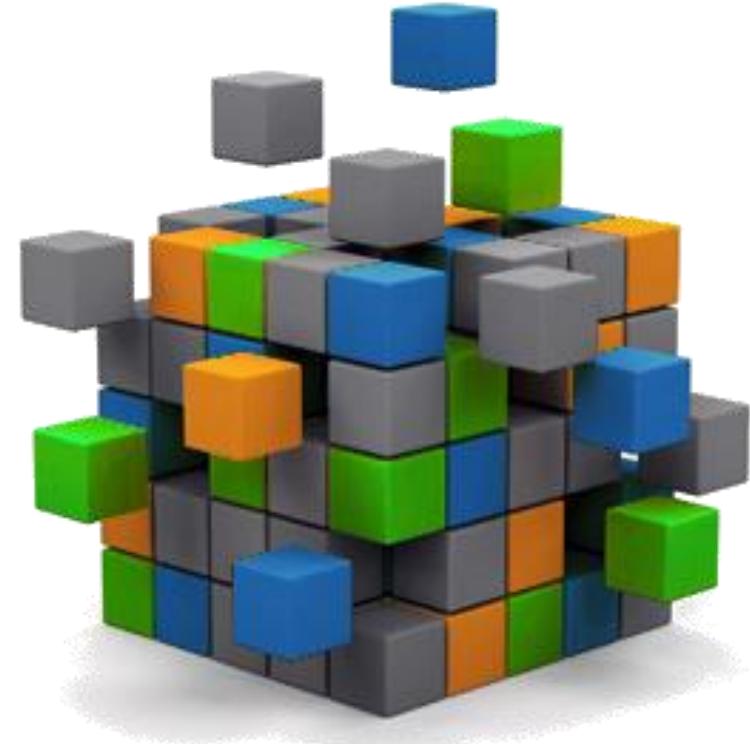
Problemlösning

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

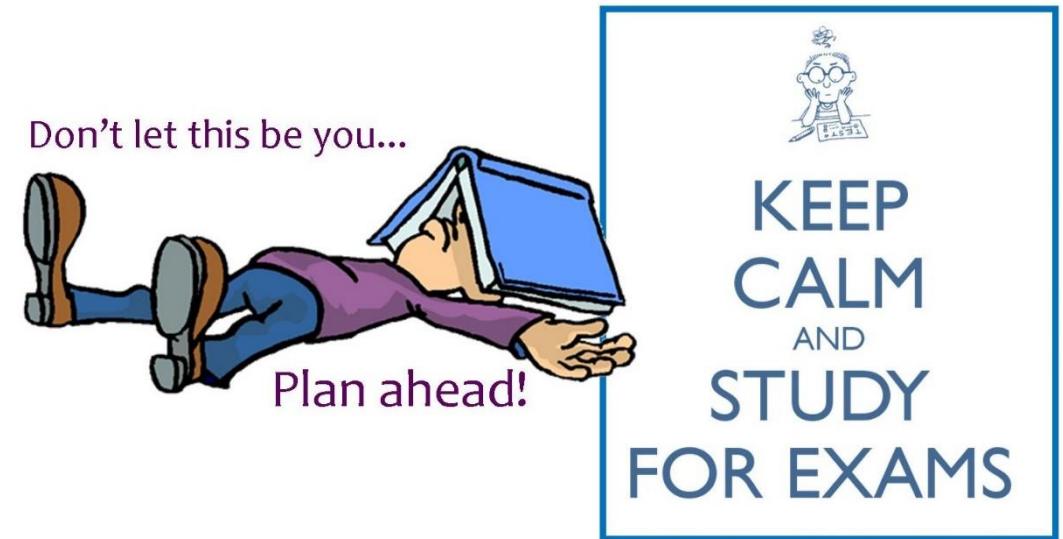
Varje modul har..

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



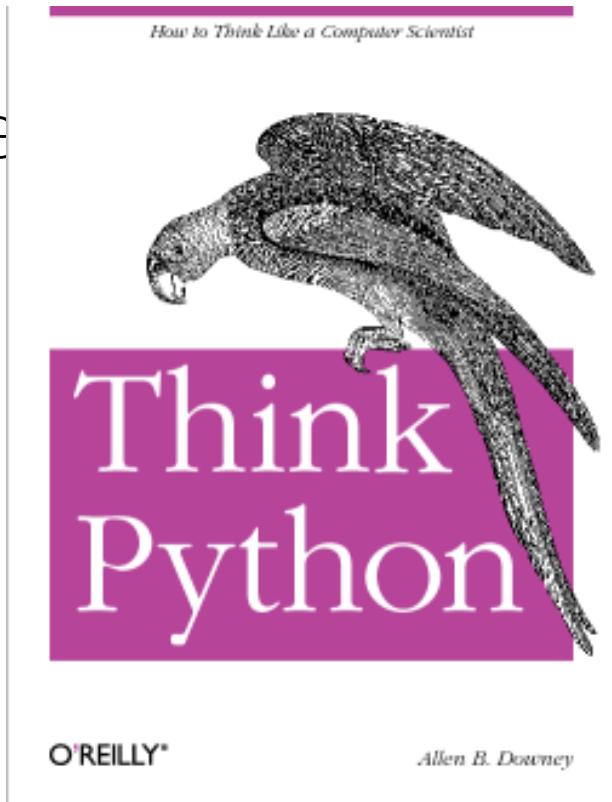
Examinerande moment

- Inlämningsuppgifter
- Deadline: söndagar varje vecka från vecka 48. Deadlines blir alltså följande:
 1. 2/12
 2. 9/12
 3. 16/12
 4. 23/12
 5. 13/1



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 motsägand 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 sträv efter att komma. (Det skadar aldrig att jobba senare delar!)

Vecka	Datum	Tid*	Vad
46	14/11	15-17	Kursintroduktion
	15/11	08-10, 10-12	Introduktion till Python + laboration
	19/11	08-12	Laboration 1: Introduktion till Python
47	21/11	13-15	Föreläsning - Funktioner
	23/11	09-10, 10-12	Inför + laboration 2: Funktioner
48	26/11	08-12	Laboration 2: Funktioner
	28/11	13-15	Föreläsning - If-satser och loopar
	29/11	09-10, 10-12	Inför laboration + laboration
49	2/12		Deadline: Inlämningsuppgift 1
	4/12	13-17	Laboration 3: If-satser och loopar
	5/12	13-15	Föreläsning - Listor och lexikon
	6/12	10-12, 13-15	Inför laboration + laboration

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.

Bra att läsa

[Think Python - Chapter 1 The way of the program:](#)

Övningar

Frågor?

