

Programming

Carsten Wulff

2010-06-15

Oversikt

- Hva er et programmeringsspråk
- Hvorfor trenger man et programmeringsspråk
- Hvordan ser et typisk språk ut
- Kompilering
- Hvilke språk fins i verden
- Hvordan ser et typisk språk ut
- Demonstrasjon

Hva er et programmeringsspråk?

Ett programmeringsspråk er et kunstig, veldefinert språk

- Et endelig antall nøkkelord (if, while, true, false)
- Et endelig antall operatorer (* + - / ! == >= <=)
- Et endelig antall blokkdefinatorer ({ } () ;)

Kan entydig forklare en datamaskin hva som skal gjøres

Hvorfor trenger man et programmeringsspråk?

- En datamaskin har et veldig begrenset vokabular (noen hundre ord), som er meget vanskelig å skrive store programmer i.
- Dette vokabularet kalles et “instruksjonssett”, og det varierer fra maskin til maskin
- Man kan skrive programmer med instruksjonssettet direkte, men det er ikke lett.
- Program skrevet direkte i et instruksjonssett vil ikke fungere på en annen maskin

Nivå 0: Maskinkode

- Bare tall
- Helt håpløst å tolke
- Men, en datamaskin må ha programmene på dette formatet for å kunne kjøre det
- Alle programmeringsspråk blir til slutt gjort om til maskinkode

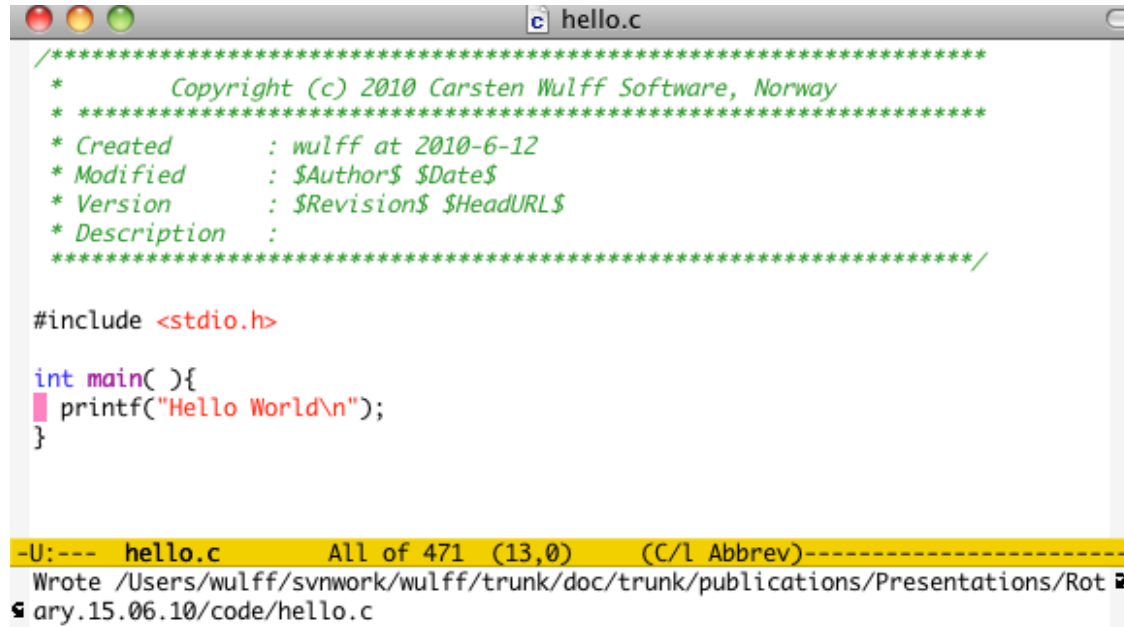
```
87654321 0011 2233 4455 6677 8899 aabb ccdd eeff 0123456789abcdef
0000e10: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e20: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e30: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e40: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e50: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e60: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e70: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e80: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000e90: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000ea0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000eb0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000ec0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000ed0: 0000 0000 6a00 4889 e548 83e4 f048 8b7d ...j.H..H..H.}
0000ee0: 0848 8d75 1089 fa83 c201 c1e2 0348 01f2 ..H.u.....H..
0000ef0: 4889 d1eb 0448 83c1 0848 8339 0075 f648 H...H...H.9.u.H
0000f00: 83c1 08e8 0800 0000 89c7 e813 0000 00f4 .....
0000f10: 5548 89e5 488d 3d13 0000 00e8 0800 0000 UH..H.=.....
0000f20: c9c3 ff25 1001 0000 ff25 1201 0000 4865 ...%...%...He
0000f30: 6c6c 6f20 576f 726c 6400 4c8d 1def 0000 llo World.L....
0000f40: 0041 53ff 25df 0000 0090 680c 0000 00e9 .AS.%....h....
0000f50: e6ff ffff 6800 0000 00e9 dcff ffff 0000 ...h.....
0000f60: 0100 0000 1c00 0000 0100 0000 2000 0000 .....
0000f70: 0000 0000 2000 0000 0200 0000 0000 0000 .....
0000f80: 0000 0000 3800 0000 3800 0000 0110 0000 ...8...8.....
0000f90: 0000 0000 3800 0000 0300 0000 0c00 0300 ...8.....
0000fa0: 1800 0100 0000 0000 100f 0001 220f 0000 .....
0000fb0: 0000 0001 0000 0000 1400 0000 0000 0000 .....
0000fc0: 017a 5200 0178 1001 100c 0708 9001 0000 .zR..x.....
0000fd0: 2c00 0000 1c00 0000 38ff ffff ffff ffff ,.....8.....
0000fe0: 1200 0000 0000 0000 0004 0100 0000 0e10 .....
0000ff0: 8602 0403 0000 000d 0600 0000 0000 0000 .....
0001000: 0000 0000 0100 0000 4810 0000 0100 0000 .....H.....
0001010: 5010 0000 0100 0000 5810 0000 0100 0000 P.....X.....
0001020: 6010 0000 0100 0000 0000 0000 0000 0000 `.....
0001030: 0000 0000 0000 0000 540f 0000 0100 0000 .....T.....
0001040: 4a0f 0000 0100 0000 0000 0000 0000 0000 J.....
0001050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0001060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0001070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0001080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0001090: 0000 0000 0000 0000 0000 0000 0000 0000 .....
--:--- a.out 42% of 37k (226,0) (HexL)-----
```

Nivå 1: Assembly code

```
hello.s
.cstring
LC0:
.asciiz "Hello World\0"
.text
.globl _main
_main:
LFB3:
pushq %rbp
LCFI0:
movq %rsp, %rbp
LCFI1:
leaq LC0(%rip), %rdi
call _puts
leave
ret
LFE3:
.section __TEXT,__eh_frame,coalesced,no_toc+strip_static_syms+live_support
EH_frame1:
.set L$set$0,LECIE1-LSCIE1
.long L$set$0
LSCIE1:
.long 0x0
.byte 0x1
.asciiz "zR\0"
.byte 0x1
.byte 0x78
.byte 0x10
.byte 0x1
.byte 0x10
.byte 0xc
.byte 0x7
.byte 0x8
.byte 0x90
.byte 0x1
.align 3
LECIE1:
.globl _main.eh
_main.eh:
LSFDE1:
.set L$set$1,LEFDE1-LASFDE1
.long L$set$1
LASFDE1:
--**-- hello.s      Top of 867 (1.32)      (Assembler)-----
```

- Rimelig vanskelig å forstå
- Beskriver hva datamaskina skal gjøre i detail
- Noen galinger skriver programmer i “assembly” for å få det til å gå raskt (for eksempel data spill)

Nivå 3: Et programmeringsspråk (C)



```
hello.c
/*****
 *      Copyright (c) 2010 Carsten Wulff Software, Norway
 *      *****/
* Created      : wulff at 2010-6-12
* Modified     : $Author$ $Date$
* Version      : $Revision$ $HeadURL$
* Description  :
*****/

#include <stdio.h>

int main( ){
    printf("Hello World\n");
}

-U:--- hello.c      All of 471 (13,0)      (C/l Abbrev)-----
Wrote /Users/wulff/svnwork/wulff/trunk/doc/trunk/publications/Presentations/Rot
ary.15.06.10/code/hello.c
```

- En smule mer forståelig
- Man bruker kodeord (int) og funksjoner (printf) for å gjøre det man vil




Øversetting: Fra C til maskinkode















- Øversettingen kalles “kompilering”
- Man trenger en kompilator (ett program)
- For språket C kan man bruke et program som heter “gcc”, som all linux og mac PC har installert fra starten av. På windows må den etterinstalleres

Demo: hello.c

Typer programmeringsspråk

- Programmeringsspråk er laget av ingeniører
- Ingeniører liker best de har laget selv
- Derav fins det mange programmeringsspråk

-  Kan gjøre meg forstått (< 10 000 linjer kode)
-  Snakker flytende (10 000 – 50 000 linjer kode)
-  Guru (mer enn 50 000 linjer kode)

Position Jun 2010	Position Jun 2009	Delta in Position	Programming Language	Ratings Jun 2010	Delta Jun 2009	Status
1	1	=	Java 	18.033%	-2.11%	A
2	2	=	C 	17.809%	+1.03%	A
3	3	=	C++ 	10.757%	+0.16%	A
4	4	=	PHP 	8.934%	-0.74%	A
5	5	=	(Visual) Basic 	5.868%	-2.07%	A
6	7	↑	C# 	5.196%	+0.66%	A
7	6	↓	Python 	4.266%	-0.49%	A
8	9	↑	Perl 	3.200%	-0.71%	A
9	45	↑↑↑↑↑↑↑↑↑↑	Objective-C 	2.469%	+2.35%	A
10	11	↑	Delphi	2.394%	+0.21%	A
11	8	↓↓↓	JavaScript 	2.191%	-1.83%	A
12	10	↓↓	Ruby 	2.070%	-0.56%	A
13	12	↓	PL/SQL 	0.787%	-0.09%	A
14	14	=	SAS	0.703%	-0.06%	A
15	15	=	Pascal	0.702%	-0.06%	A-
16	18	↑↑	Lisp/Scheme/Clojure 	0.654%	+0.05%	B
17	19	↑↑	Lua	0.592%	+0.04%	B
18	20	↑↑	MATLAB 	0.589%	+0.06%	B
19	16	↓↓↓	ABAP	0.577%	-0.15%	B
20	27	↑↑↑↑↑↑↑↑	PowerShell	0.529%	+0.23%	B

Hva brukes språkene til

Navn	Type	Bruksområde
Java	Kompilert til et mellomformat	Web (nettbanker), GUI programmer, det meste.
C	Kompilert	Alt mulig (Spill, mobiltelefoner, vaskemaskiner, pulsklokker)
C++	Kompilert	Alt mulig (Word, Excel, Photoshop)
PHP	Interpretert	Web (Wikipedia)
(Visual) Basic	Kompilert	Makroer i Excel og Word, Windows programmer
C#	Kompilert	Windows programmer
Perl	Interpretert	Behandling av store tekstmengder. Alt mulig
Objective-C	Kompilert	iPhone og Mac
SQL	Interpretert	Henting av data fra databaser
Matlab	Interpretert	Matematiske beregninger

Demo

1. Data ut fra et simuleringsprogram
2. Perl script for parsing av data
3. C# program for plotting av data

Oppsummering

- Alt må til slutt oversettes til maskinkode, det er det eneste en datamaskin kan forstå
- Det fins mange programmeringsspråk, noe er smak og behag og noe er effektivitet
- Hvis du vil lære deg programmering er du nødt til å ha en oppgave å som du vil ha løst
- Hvis du skal lære deg programmering må du være tålmodig

Introduction to Programming Perl: Three Virtues of a Programmer

by Larry Wall (creator of Perl) et al.

- **Laziness** - The quality that makes you go to great effort to reduce overall energy expenditure. It makes you write labor-saving programs that other people will find useful, and document what you wrote so you don't have to answer so many questions about it. Hence, the first great virtue of a programmer. Also hence, this book. See also impatience and hubris.
- **Impatience** - The anger you feel when the computer is being lazy. This makes you write programs that don't just react to your needs, but actually anticipate them. Or at least pretend to. Hence, the second great virtue of a programmer. See also laziness and hubris.
- **Hubris** - Excessive pride, the sort of thing Zeus zaps you for. Also the quality that makes you write (and maintain) programs that other people won't want to say bad things about. Hence, the third great virtue of a programmer. See also laziness and impatience.