



HF0010

Introduktionskurs i Datateknik

F3: World Wide Web?



Introduktionskurs i Datateknik

F1: Vad är en dator?

F2: programmeringsspråk?

L1: Hour of code, alla kan programmera, Diplom!

L2: Visuellt programmering av svårare problem!

F3: Webbprogrammering – JavaScript!

L3: IDE, Web sida & JavaScript!

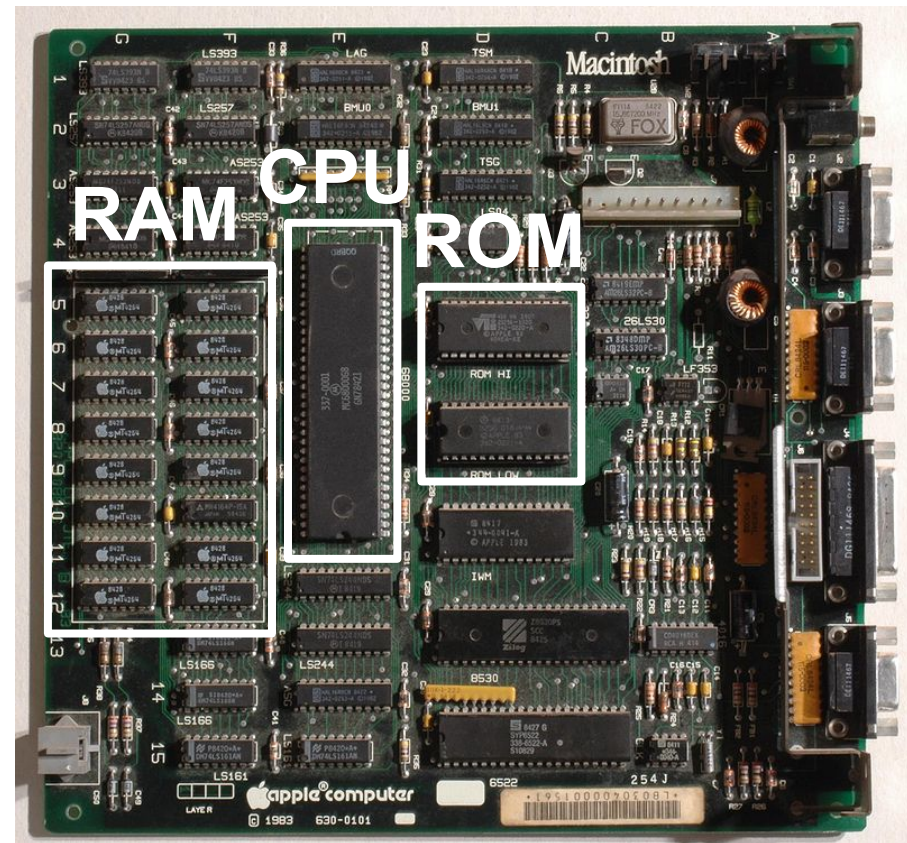
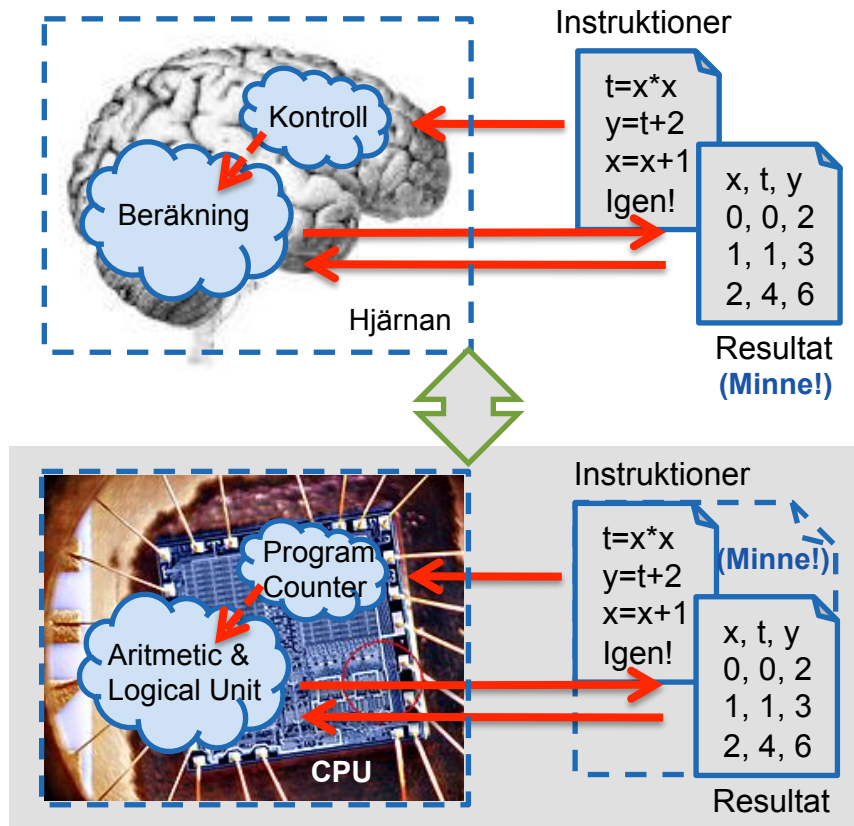


HTML



Vi ska lära oss programmera...

...då måste vi nog kunna två saker...

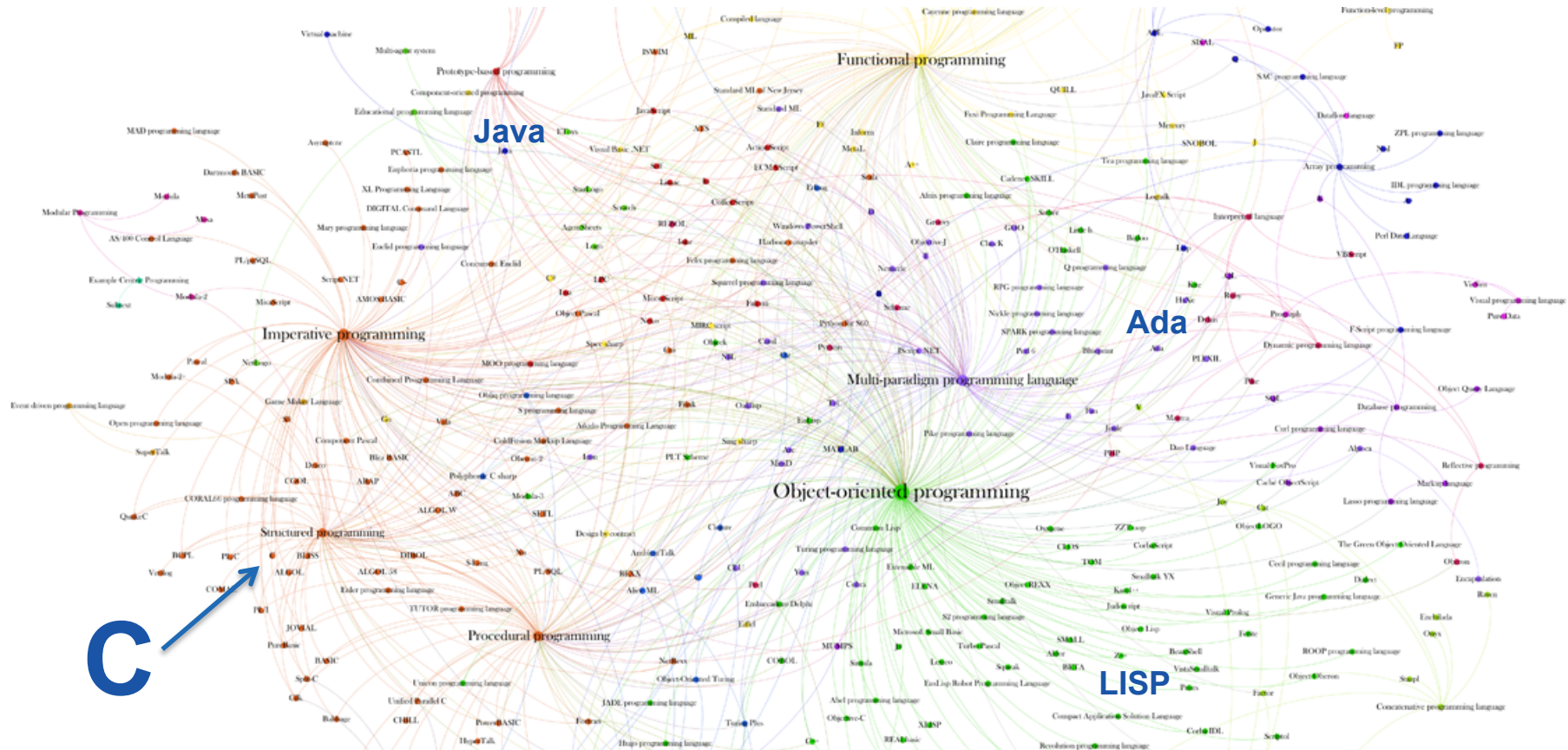


...ett programmeringsspråk och hur det används!



1b

Ja det gör det, men välj rätt!



Generell, maskinnära, processororienterad, funktionell...



1c

Syntax, semantik och reserverade ord!

```
while ( closed ( 'A' , number + 1 ) ) ;
```

Vad är "ord"
(tokens)
i språket!

Syntax

Identifiers
*[A-Za-z_][A-Za-z0-9_]**

Literals Operators

(Tokens)

Punctuation marks
,;(){}::white space

(Keywords ≈50!)

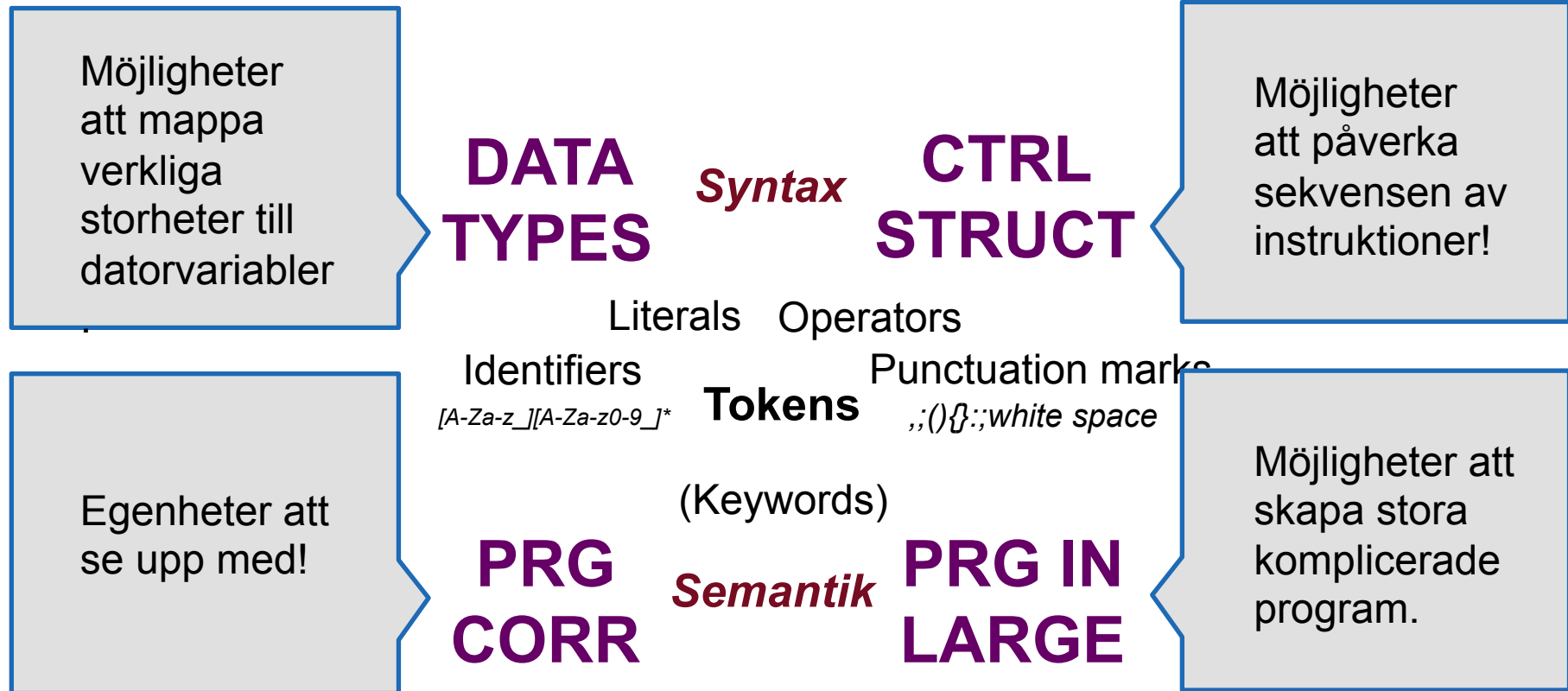
Semantik

Vad är en
mening
i språket!

Antalet mellanslag spelar nästan alltid ingen roll!



Likheter...



Vi börjar med att titta på vad data typer är...



2a

Hur använder man (dator-)språket?

Arbetsintensiv Monoton

Konsten att översätta en **verklig process** till datorns värld...

Farlig Felkänslig



Väg 4 tomater och beräkna medelvärdet!



2b

Hur använder man (dator-)språket?

Arbetsintensiv Monoton

Konsten att översätta en **verklig process** till datorns värld...

Farlig Felkänslig



Väg 4 tomater och beräkna medelvärdet!



Nollställ miniräknaren.

Repetera 4 gånger:

Väg nästa tomat...

**...addera vikten till
summan på miniräknaren.**

Dividera summan med 4!



2c

Hur använder man (dator-)språket?

Arbetsintensiv Monoton

Konsten att översätta en **verklig process** till datorns värld...

Farlig Felkänslig



➡ **Väg 4 tomater och beräkna medelvärdet!**
⬇

Nollställ miniräknaren.

Repetera 4 gånger:
Väg nästa tomat...
...addera vikten till
summan på miniräknaren.

Dividera summan med 4!

➡

```
int measurement;  
int total = 0;  
  
for (int counter=0; counter<4; counter++) {  
    scanf("%d",&measurement);  
    total+=measurement;  
}  
  
printf("Avrage = %f \n", total/4.0);
```

...det är att kunna programmera, och rätt lika från språk till språk!



Reflektioner från Lab #1...

The screenshot shows a Minecraft game window on the left with a tree in the center. On the right is a programming interface. At the top right, a text box says: "Trä är en viktig resurs. Mycket kan tillverkas av det. Gå till trädet och använd kommandot 'förstör block' för att hugga ned det." Below this is a tabbed interface with "bitar" (blocks) and "Arbetsyta:" (workspace). The "bitar" tab contains four blocks: "gå framåt", "förstör block", "sväng vänster", and "sväng höger". The "Arbetsyta:" tab shows a sequence of three blocks: "vid start", "gå framåt", and "förstör block", connected by a blue arrow pointing down to the text "Sekvens...". At the bottom right, a code block contains the following text:

```
moveForward();  
moveForward();  
destroyBlock();
```



Reflektioner från Lab #1...



Varje hus börjar med en vägg. Bygg den första delen av ditt hus genom att lägga "placera"- och "gå framåt"-kommandona i upprepningsloopen.

bitar

Arbetsyta: Börja om Visa kod

gå framåt

placera björkplankor ▾

sväng vänster ↶ ▾

sväng höger ↷ ▾

upprepa 4 ▾ gånger
gör

vid start

upprepa 4 ▾ gånger
gör
placera björkplankor ▾
gå framåt

Iteration...

```
for (var count = 0; count < 4; count++) {  
  placeBlock("planksBirch");  
  moveForward();  
}
```

4ggr



Reflektioner från Lab #1...



Kör

Det är bra att planera framåt. Plantera grödor på båda sidor om vattnet så att du inte blir hungrig längre fram.

bitar

Arbetsyta: 12 / 11 block

Börja om Visa kod

gå framåt

plantera gröda

sväng vänster ↶

sväng höger ↷

upprepa 6 gånger

gör

sax

vid start

upprepa 6 gånger

gör

plantera gröda

gå framåt

sväng höger ↷

gå framåt

gå framåt

sväng höger ↷

gå framåt

upprepa 6 gånger

gör

plantera gröda

gå framåt



Reflektioner från Lab #1...

Det är bra att planera framåt. Plantera grödor på båda sidor om vattnet så att du inte blir hungrig längre fram.

```
for (var count2 = 0; count2 < 2; count2++) {  
  for (var count = 0; count < 6; count++) {  
    plantCrop();  
    moveForward();  
  }  
  turnRight();  
  moveForward();  
  moveForward();  
  turnRight();  
  moveForward();  
}
```

Arbetsyta: Börja om Visa kod

vid start
upprepa 2 gånger
gör
 upprepa 6 gånger
 gör
 plantera gröda
 gå framåt
sväng höger
gå framåt
gå framåt
sväng höger
gå framåt

Nästlade iterationer!

Reflektioner från Lab #1...

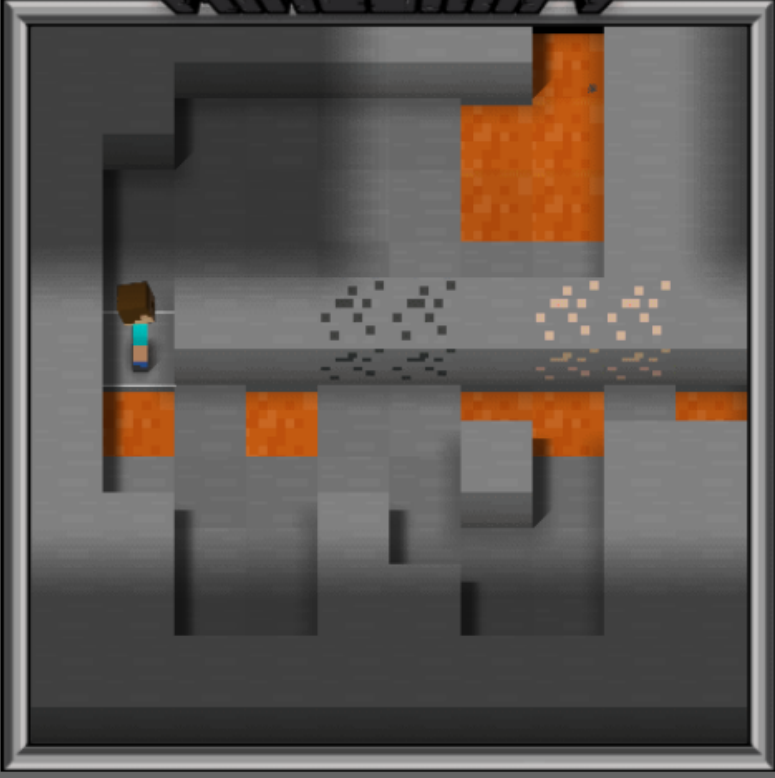


The screenshot shows a Minecraft game window on the left with a character in a cave. On the right is a custom programming interface. At the top right, a text box explains: "Lava döljer sig under några av de här blocken, och du måste täcka den innan du kan gå framåt. Ett 'om'-kommando blir praktiskt här. Lägg till ett 'gå framåt'-kommando på rätt plats för att bryta de här blocken." Below this is a command palette with "bitar" (bits) and "Arbetsyta: 6 / 6 block". A list of actions includes "gå framåt", "sväng vänster", "sväng höger", "förstör block", "upprepa 3", "om lava fram", and "utför". A code editor on the right shows a block-based script: "vid start", "upprepa 7 gånger", "gör", "förstör block", "om lava framför", "utför", "placera kullersten framåt", and "gå framåt". A callout box labeled "Selektion" displays the corresponding JavaScript code:

```
for (var count = 0; count < 7; count++) {  
  destroyBlock();  
  ifLavaAhead(function() {  
    placeBlockAhead("cobblestone");  
  });  
  moveForward();  
}
```




Reflektioner från Lab #1...



Lava döljer sig under några av de här blocken, och du måste täcka den innan du kan gå framåt. Ett "om"-kommando blir praktiskt här. Lägg till ett "gå framåt"-kommando på rätt plats för att bryta de här blocken.

bitar

Arbetsyta: 6 / 6 block Börja om Visa kod

gå framåt

sväng vänster

sväng höger

förstör block

upprepa 3 gånger

om lava framför

vid start

upprepa 7 gånger

gör

förstör block

om lava framför

utför

placera kullersten framåt

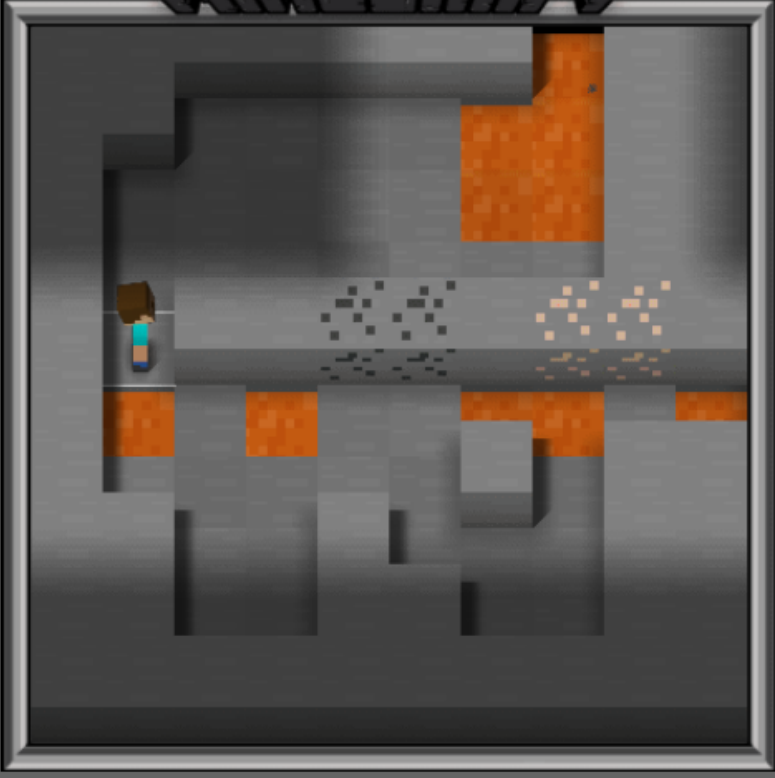
gå framåt

Sekvens Selektion

```
for (var count = 0; count < 7; count++) {  
  destroyBlock();  
  ifLavaAhead(function() {  
    placeBlockAhead("cobblestone");  
  });  
  moveForward();  
}
```



Reflektioner från Lab #1...



MINECRAFT

Lava döljer sig under några av de här blocken, och du måste täcka den innan du kan gå framåt. Ett "om"-kommando blir praktiskt här. Lägg till ett "gå framåt"-kommando på rätt plats för att bryta de här blocken.

bitar

Arbetsyta: 6 / 6 block Börja om Visa kod

gå framåt

sväng vänster

sväng höger

förstör block

upprepa 3 gånger

om lava framför

vid start

upprepa 7 gånger

gör

förstör block

om lava framför

utför

placera kullersten framåt

gå framåt

Iteration

Sekvens

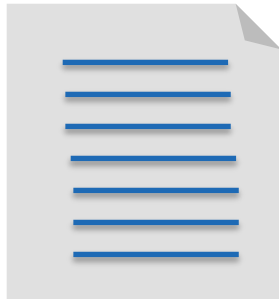
Selektion

```
for (var count = 0; count < 7; count++) {
  destroyBlock();
  ifLavaAhead(function() {
    placeBlockAhead("cobblestone");
  });
  moveForward();
}
```

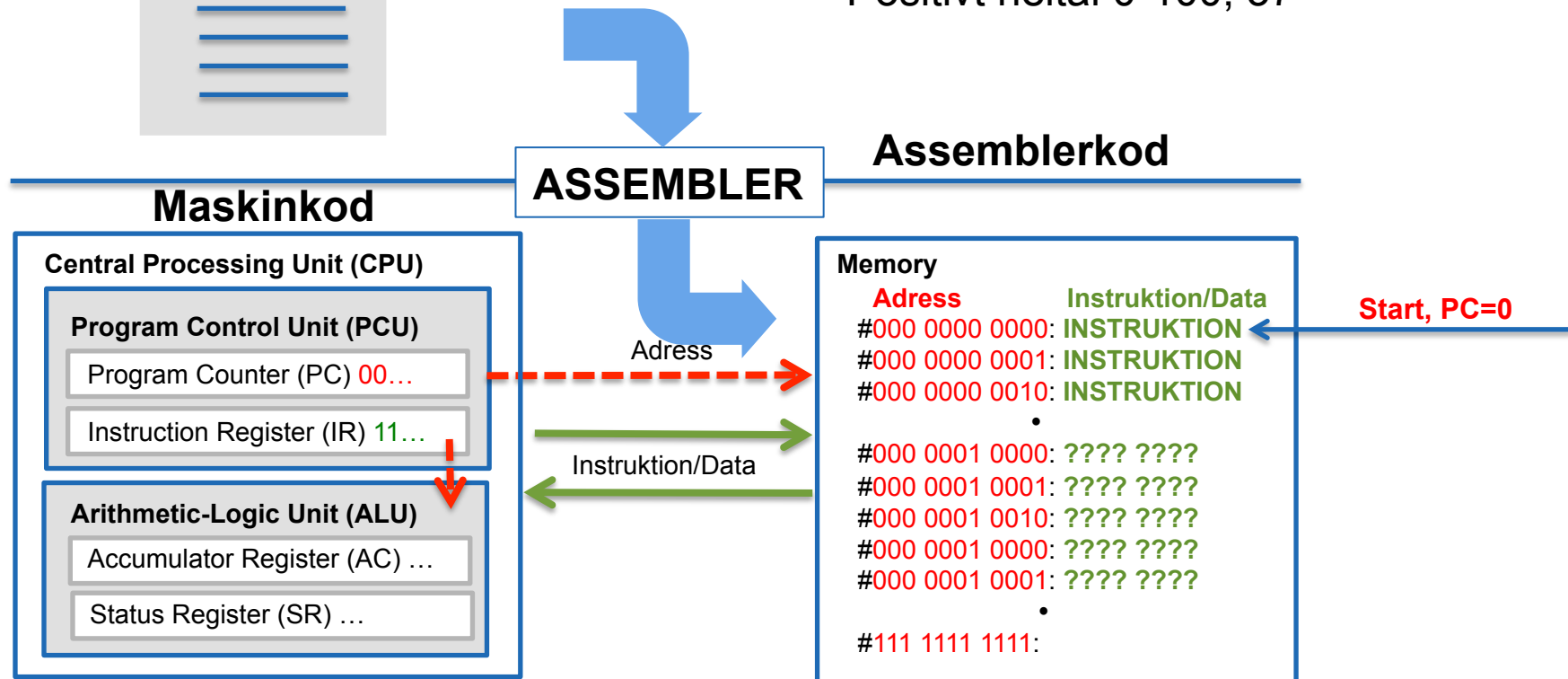


Järnkoll?

Feberdatalogger!



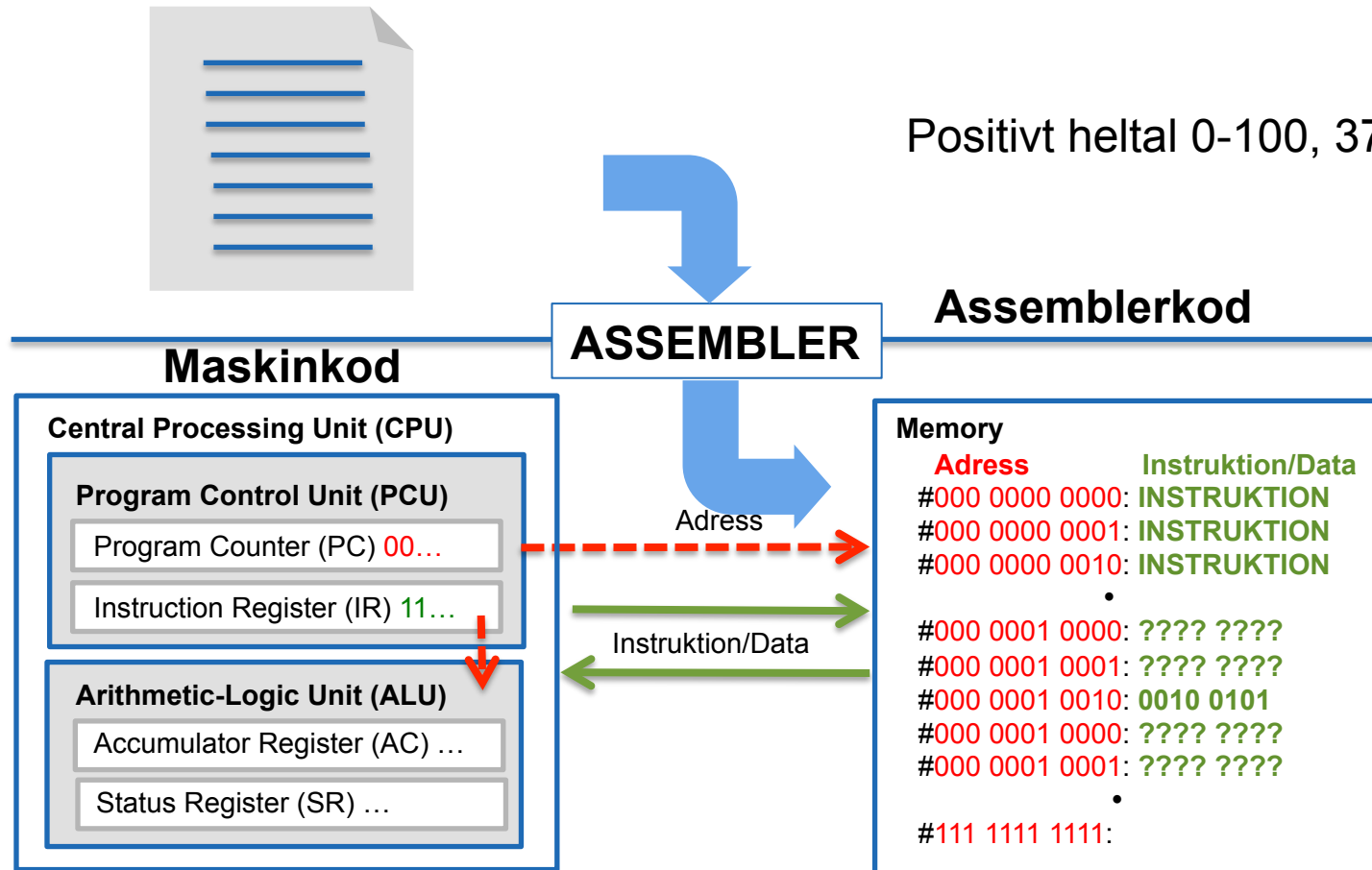
Positivt heltal 0-100, 37





Järnkoll?

Feberdatalogger!



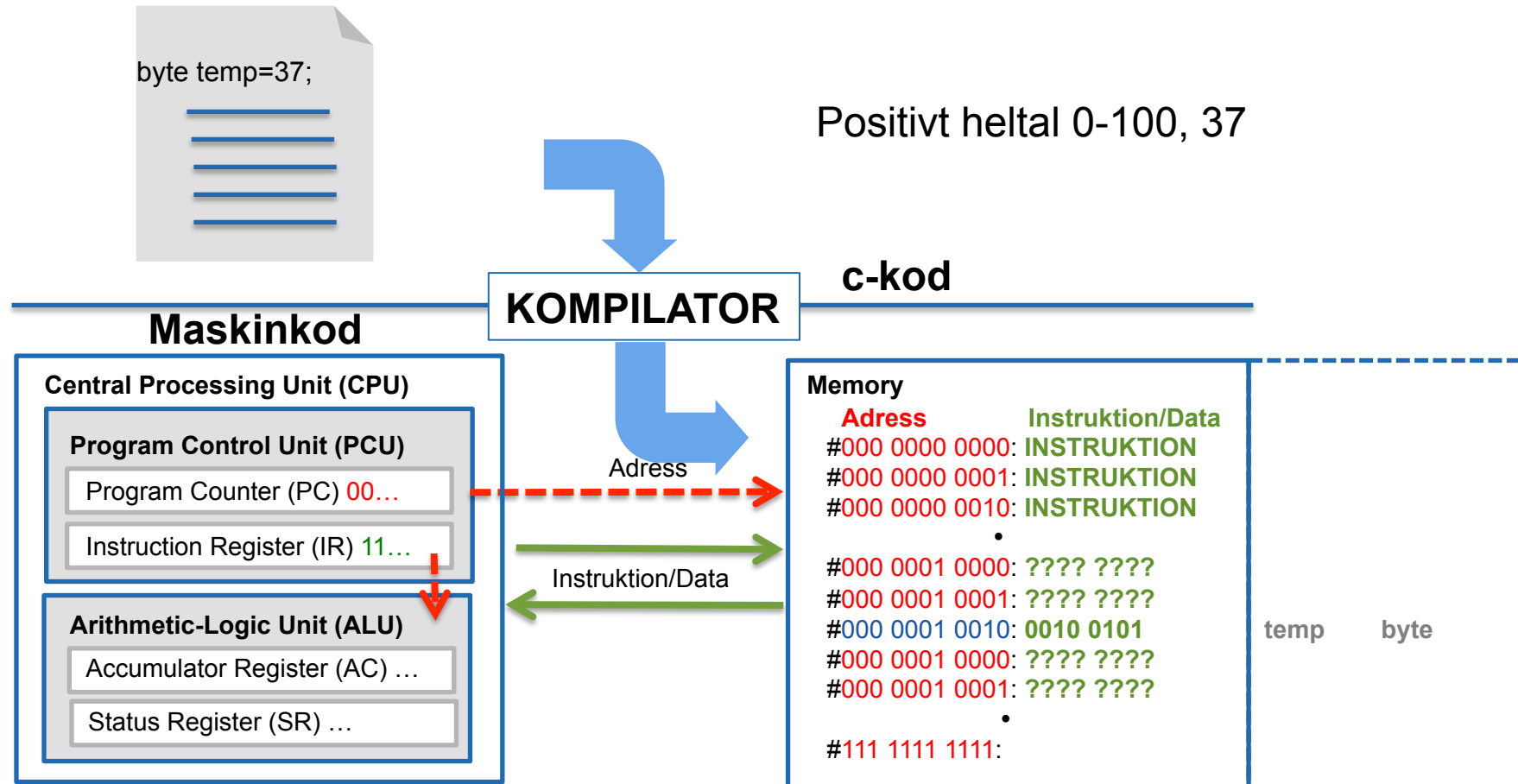
Positivt heltal 0-100, 37 0010 0101

”Vi” måste komma ihåg att temperaturen finns lagrad i minnes-cell 34 och att bitmönstret ska tolkas som ett positivt heltal!



Järnkoll?

Feberdatalogger!

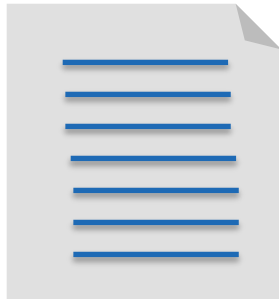


Kompilator väljer en ledig minnescell som "vi" därefter kan nå via variabelnamnet temp, för tal som är mellan 0 och 255.

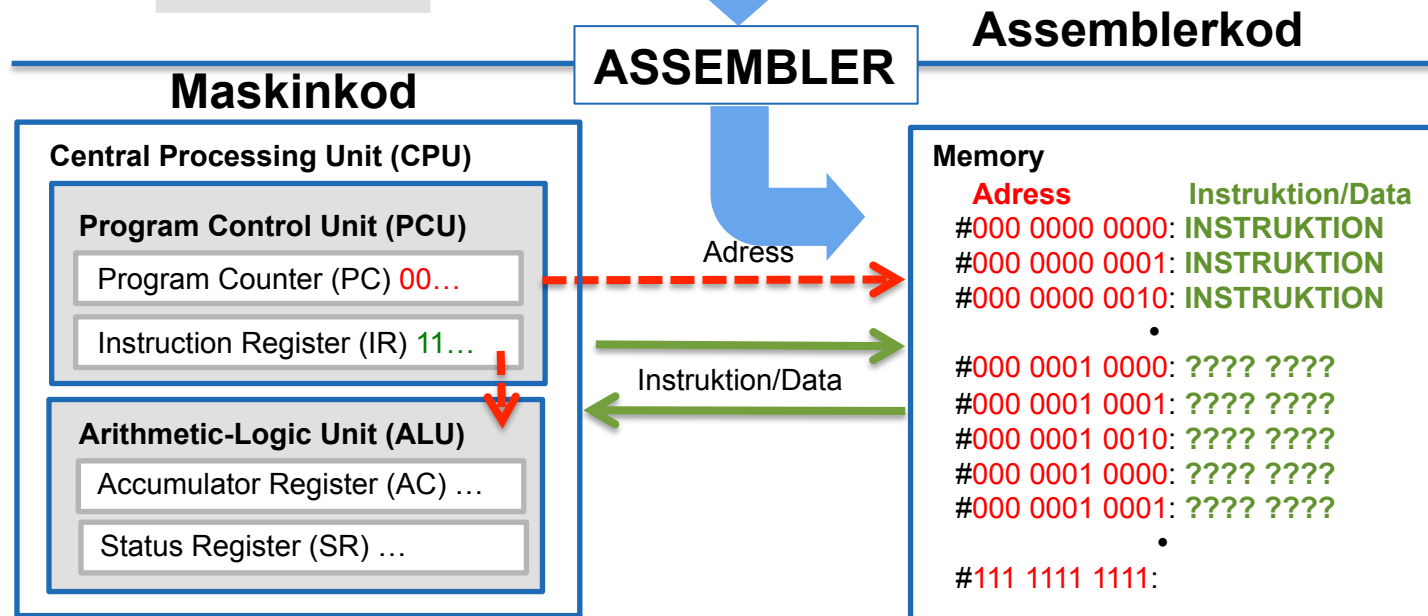


Järnkoll?

Tempdatalogger!



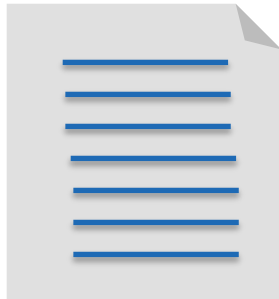
Positivt heltal 0-1000, 800





Järnkoll?

Tempdatalogger!



ASSEMBLER

Positivt heltal 0-1000, 800

0000 0011 0010 0000

512
256

Max 255

Assemblerkod

Maskinkod

Central Processing Unit (CPU)

Program Control Unit (PCU)

Program Counter (PC) 00...

Instruction Register (IR) 11...

Arithmetic-Logic Unit (ALU)

Accumulator Register (AC) ...

Status Register (SR) ...

Adress

Instruktion/Data

Memory

Adress **Instruktion/Data**

#000 0000 0000: INSTRUKTION

#000 0000 0001: INSTRUKTION

#000 0000 0010: INSTRUKTION

.

#000 0001 0000: ???? ????

#000 0001 0001: ???? ????

#000 0001 0010: 0010 0000 (LSB)

#000 0001 0000: 0000 0011 (MSB)

#000 0001 0001: ???? ????

.

#111 1111 1111:

Little-endian vs Big-endian





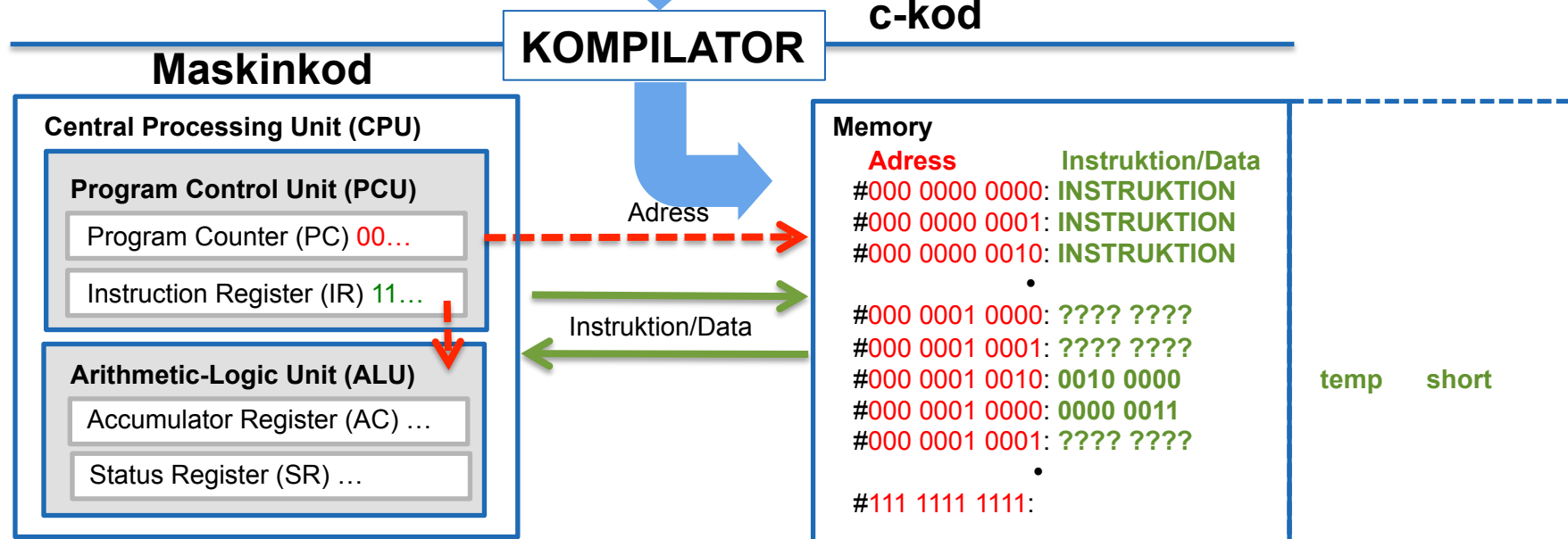
Järnkoll?

Tempdatalogger!

short temp=800;

Positivt heltal 0-1000, 800

0000 0011 0010 0000



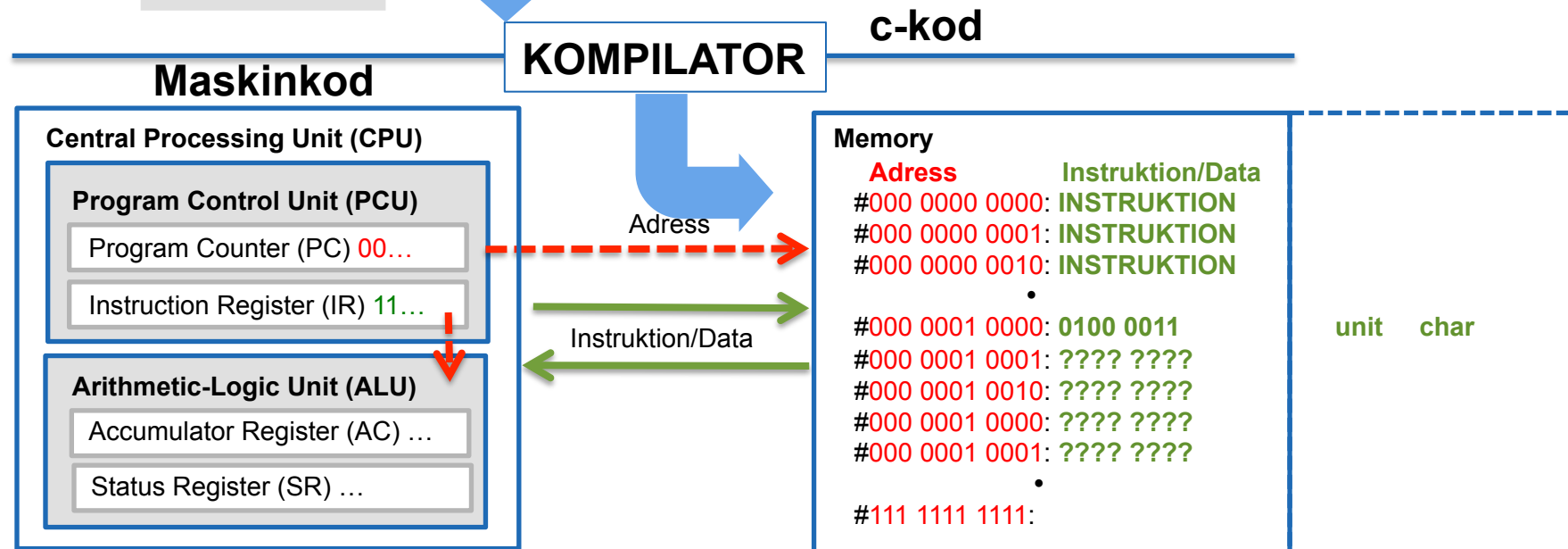


Bokstäver?! Järnkoll?

char unit='C';

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL



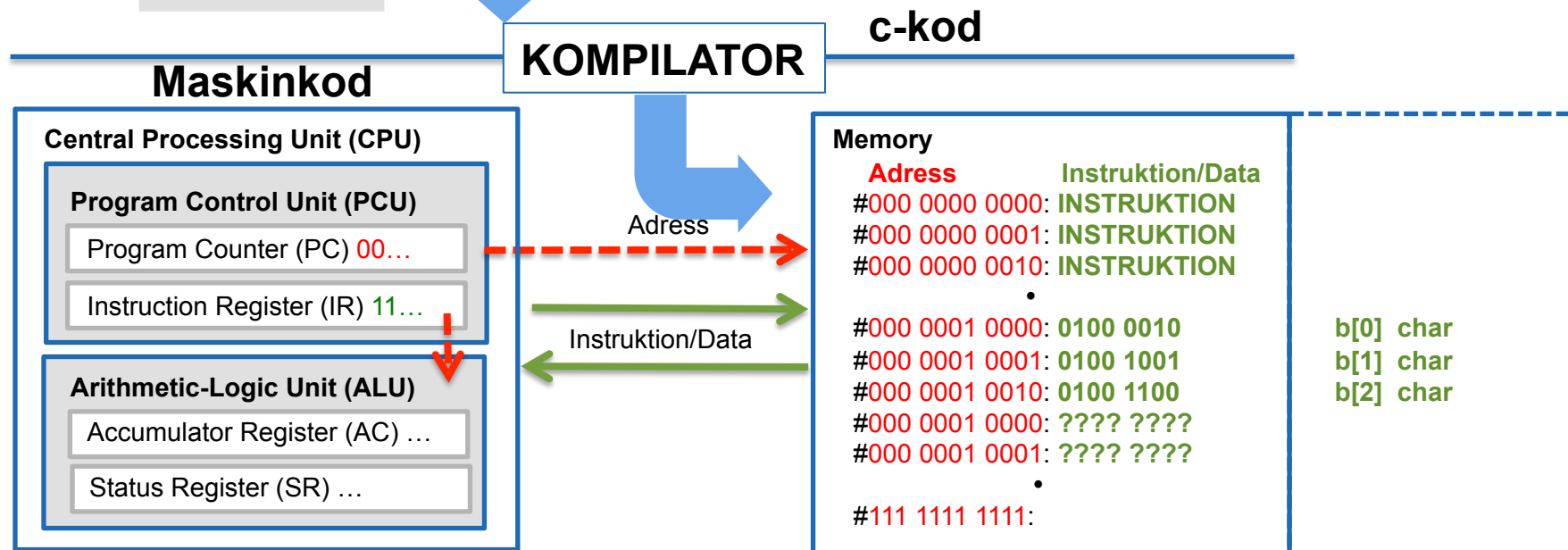


Bokstäver?! Järnkoll?

```
char b[]="BIL";
```

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL





Järnkoll?

```
struct id{ char fst,  
            char snd,  
            char trd,  
            short nb  
} car = {'b','i','l',1};
```

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

KOMPILATOR

c-kod

Maskinkod

Central Processing Unit (CPU)

Program Control Unit (PCU)

Program Counter (PC) 00...

Instruction Register (IR) 11...

Arithmetic-Logic Unit (ALU)

Accumulator Register (AC) ...

Status Register (SR) ...

Memory

Address Instruktion/Data

#000 0000 0000: INSTRUKTION

#000 0000 0001: INSTRUKTION

#000 0000 0010: INSTRUKTION

.

#000 0001 0000: 0100 0010

#000 0001 0001: 0100 1001

#000 0001 0010: 0100 1100

#000 0001 0000: 0000 0001

#000 0001 0001: 0000 0000

.

#111 1111 1111:

fst char
snd char
trd char
nb short

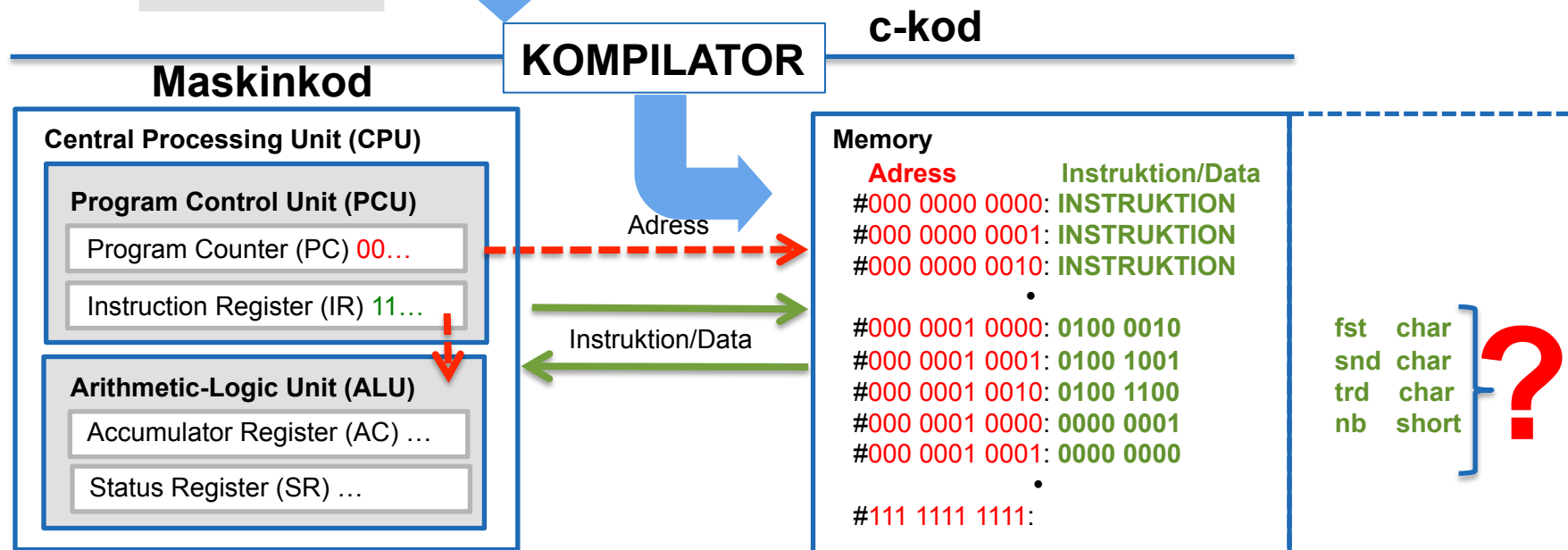




Järnkoll?

```
struct id{ char fst,  
            char snd,  
            char trd,  
            short nb  
} car = {'b','i','l',1};
```

B I L 0 0 1



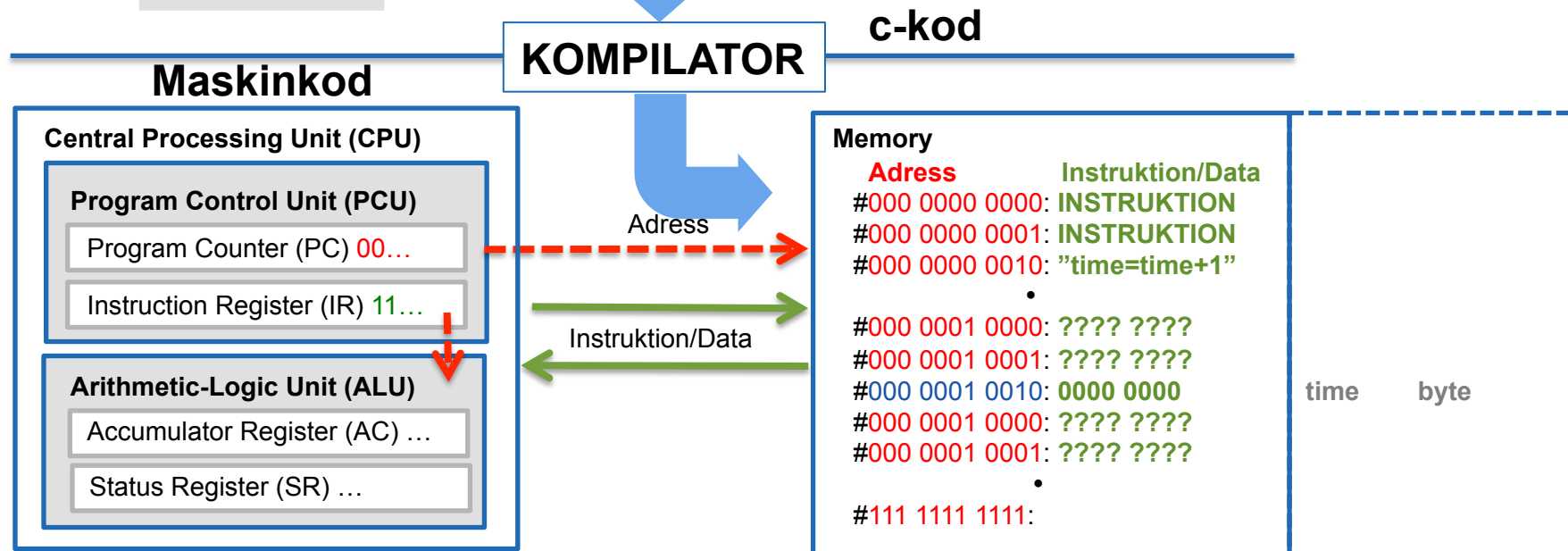
Det är oerhört värdefullt om det går att skapa variabler som efterliknar verkliga storheter...



Järnkoll?

```
byte time=0;  
_____  
time=time+1;  
_____
```

time=time+1;



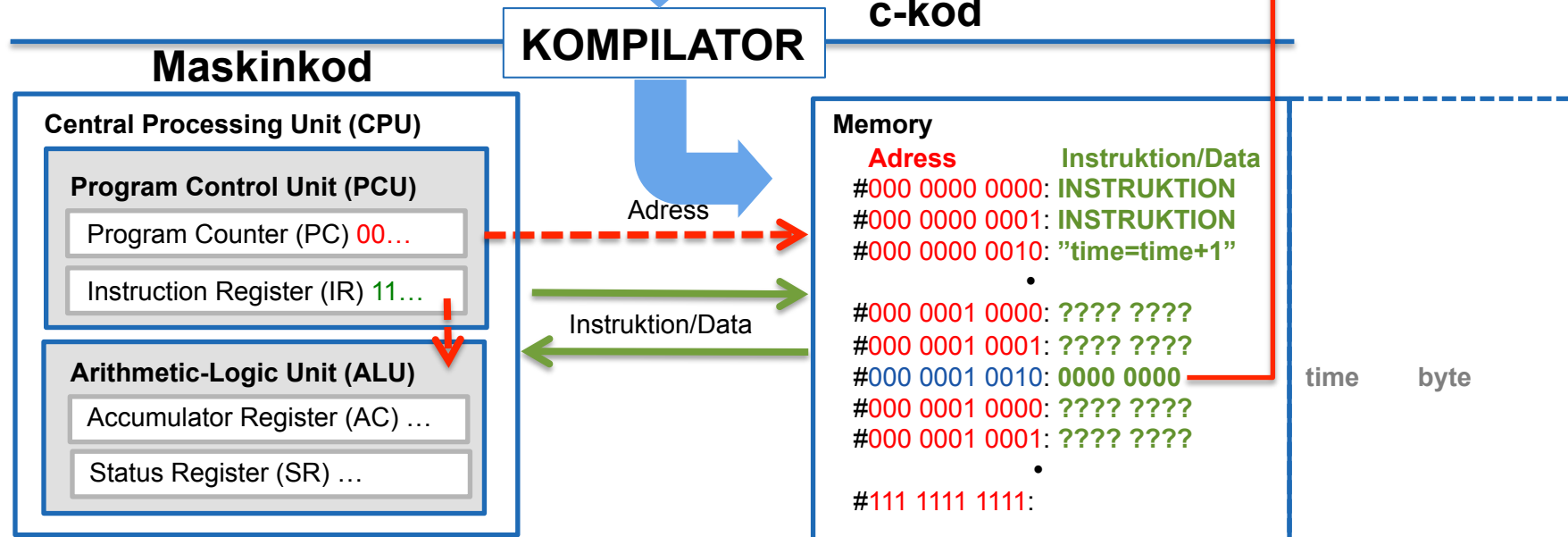


Järnkoll?

```
byte time=0;
```

```
time=time+1;
```

time= 0 +1;



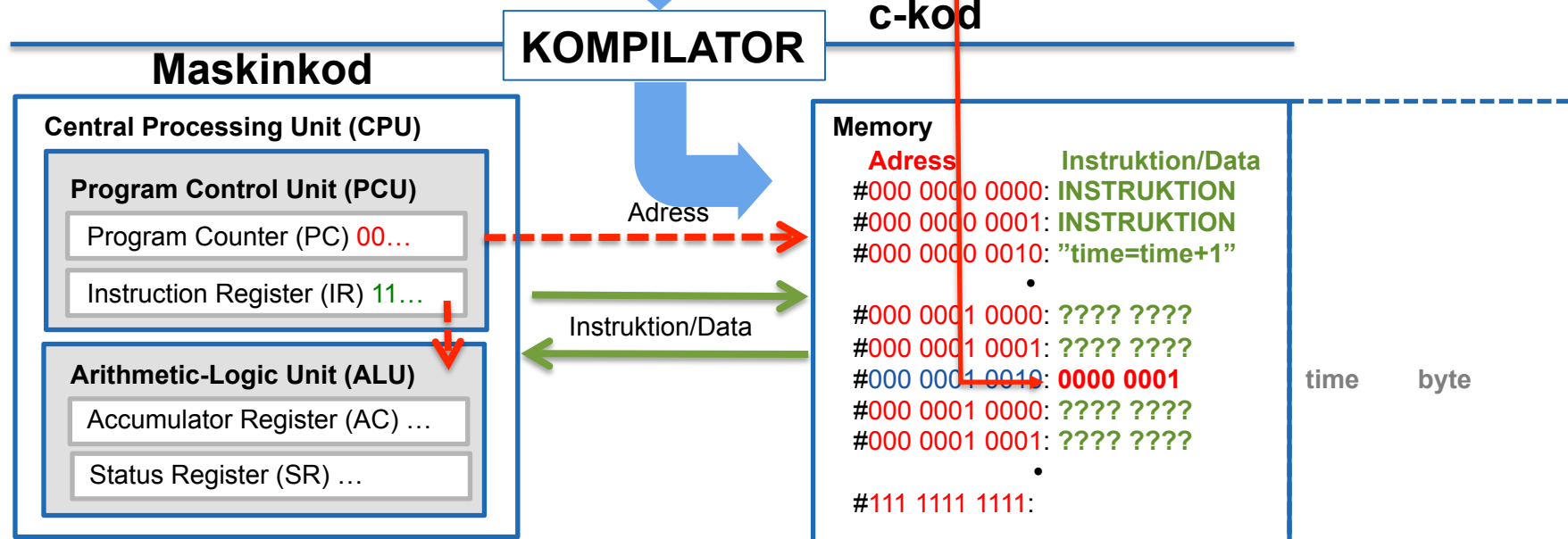


Järnkoll?

```
byte time=0;
```

```
time=time+1;
```

1 = 0 + 1;



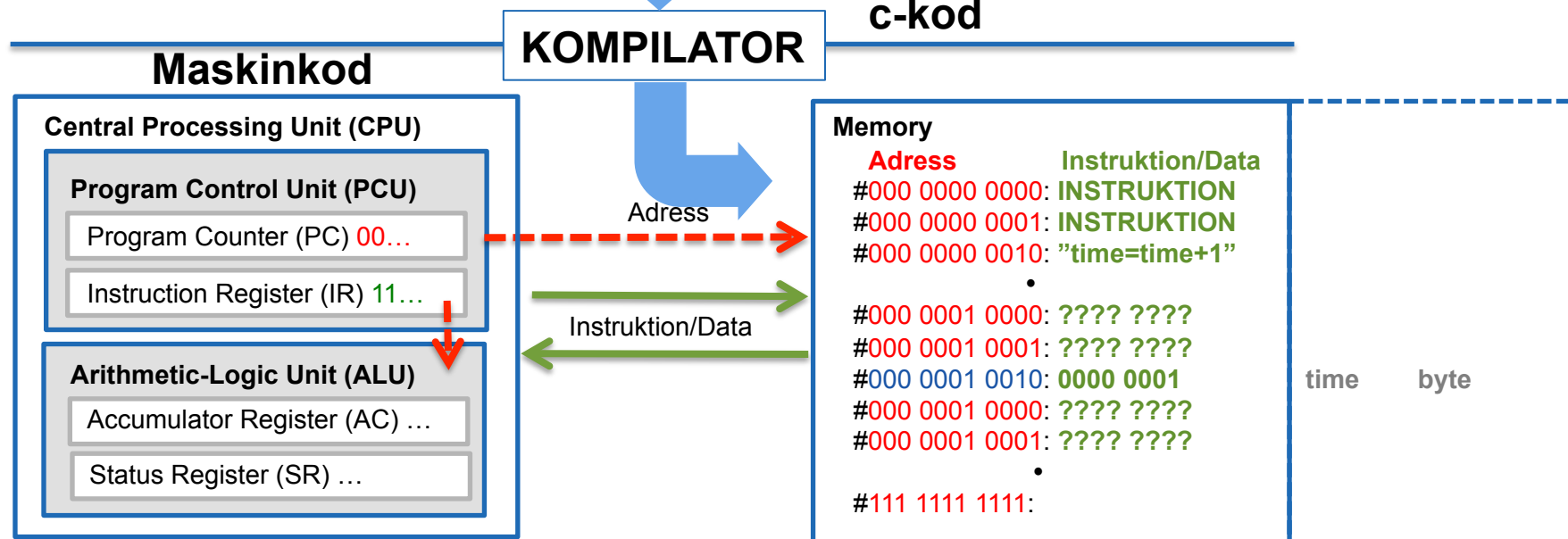


Järnkoll?

```
byte time=0;
```

```
time=time+1;
```

time=time+1;



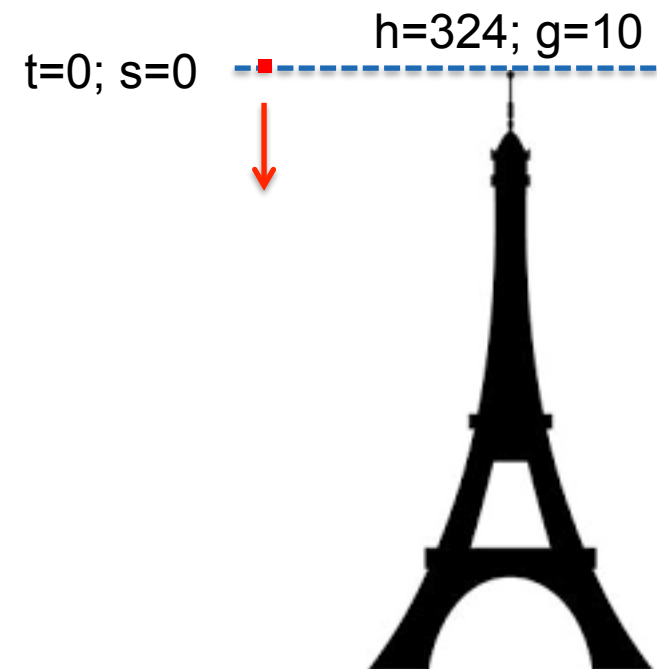
$time_{new} = time_{old} + 1;$
 $time_{n+1} = time_n + 1;$



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \quad s=0.5at^2$$





Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

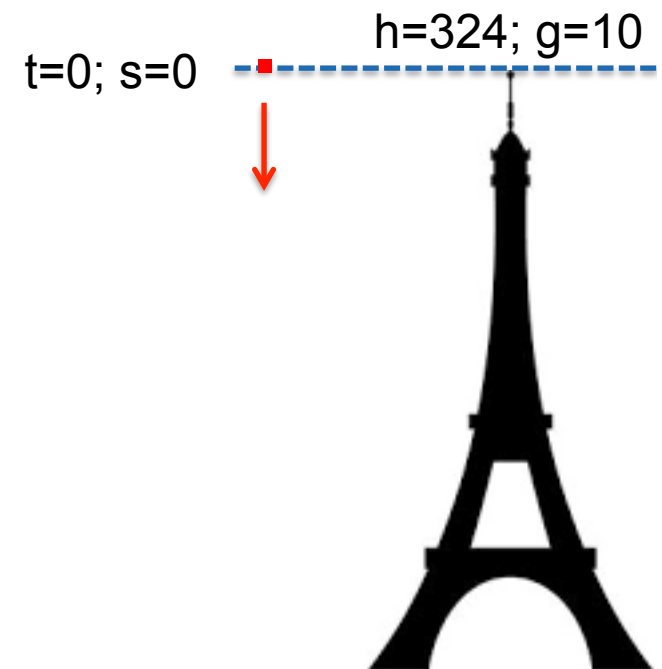
t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

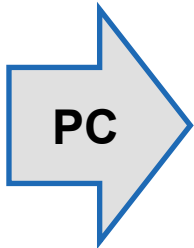
Visa klart!





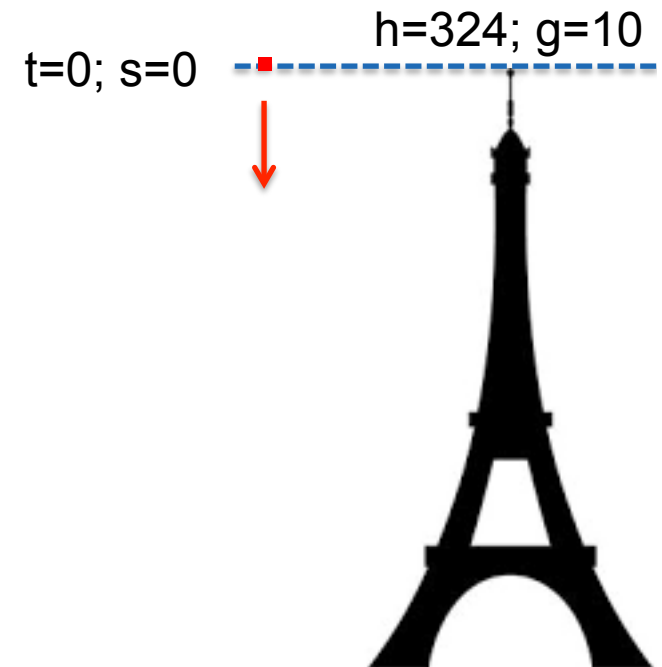
Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!



t=0, s=0, v=0
Visa programtitel
Visa tabellrubrik
Upprepa...
 visa t, s & v
 t = t + 1
 v = 10*t
 s = 0.5*10*t*t
...så länge s<324
Visa klart!

$$v=at \quad \& \\ s=0.5at^2$$



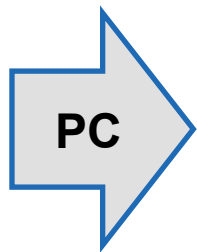
Minne

324: 0 (t heltal)
325: 0 (s heltal)
326: 0 (v heltal)



Järnkoll?

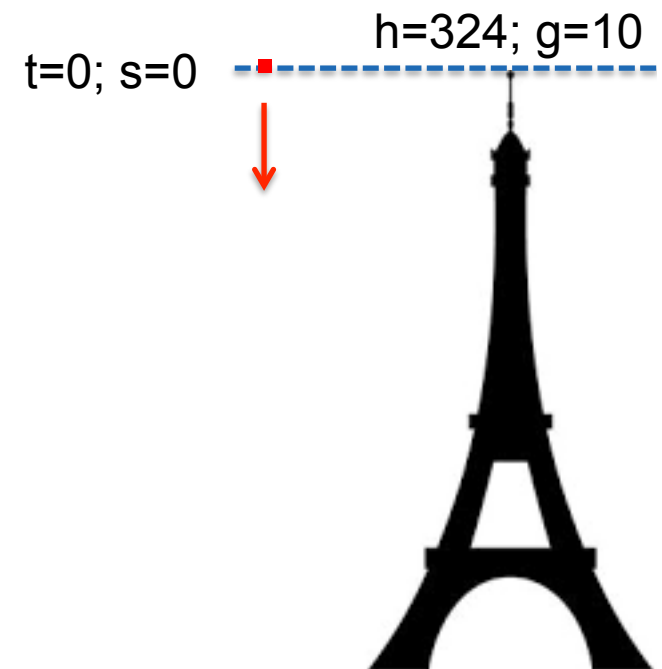
Tabell för v & s för varje sekund tills bollen träffar marken!



t=0, s=0, v=0
Visa programtitel
Visa tabellrubrik
Upprepa...
 visa t, s & v
 t = t + 1
 v = 10*t
 s = 0.5*10*t*t
...så länge s<324
Visa klart!

Linjär acc!!!

$$v=at \quad \&$$
$$s=0.5at^2$$



Minne

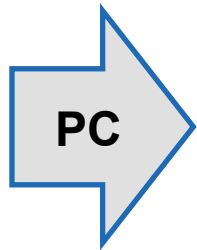
324: 0 (t heltal)
325: 0 (s heltal)
326: 0 (v heltal)



Järnkoll?

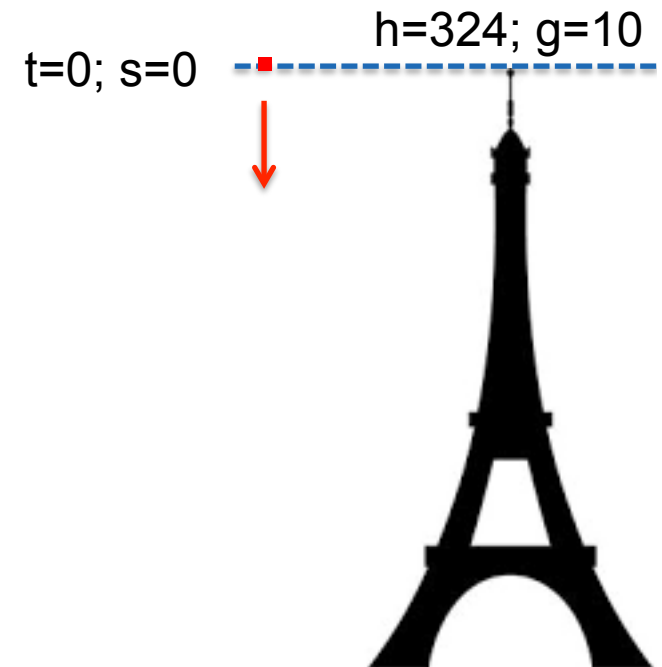
Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$



t=0, s=0, v=0
Visa programtitel
Visa tabellrubrik
Upprepa...
 visa t, s & v
 t = t + 1
 v = 10*t
 s = 0.5*10*t*t
...så länge s<324
Visa klart!

Linjär acc!!!
t v s



Minne

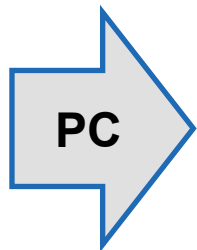
324: 0 (t heltal)
325: 0 (s heltal)
326: 0 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$



t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

Linjär acc!!!

t v s

Uppfattat!

t=0; s=0

h=324; g=10



Minne

324: 0 (t heltal)

325: 0 (s heltal)

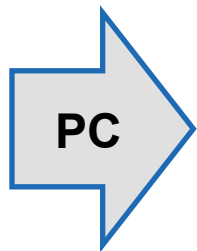
326: 0 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$



t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

Linjär acc!!!

t	v	s
---	---	---

0	0	0
---	---	---

t=0; s=0

h=324; g=10



Minne

324: 0 (t heltal)

325: 0 (s heltal)

326: 0 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

Linjär acc!!!

t	v	s
---	---	---

0	0	0
---	---	---

t=0; s=0

h=324; g=10



PC

Minne

324: 1 (t heltal)

325: 0 (s heltal)

326: 0 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

PC

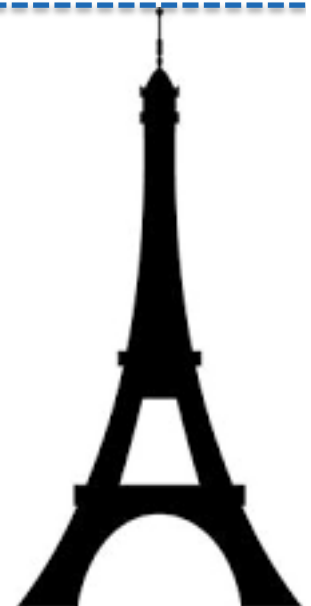
Linjär acc!!!

t	v	s
---	---	---

0	0	0
---	---	---

t=0; s=0

h=324; g=10



Minne

324: 1 (t heltal)

325: 0 (s heltal)

326: 10 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

PC

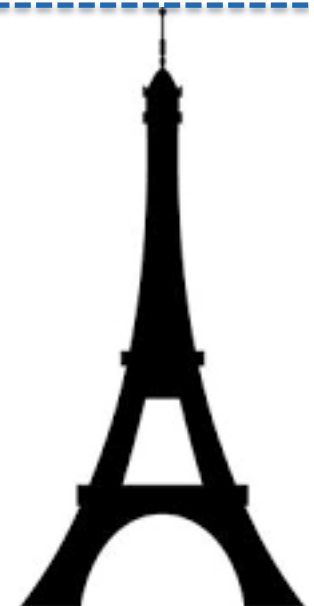
Linjär acc!!!

t	v	s
---	---	---

0	0	0
---	---	---

t=0; s=0

h=324; g=10



Minne

324: 1 (t heltal)

325: 5 (s heltal)

326: 10 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
---	---	---

0	0	0
---	---	---

t=0; s=0

h=324; g=10



SANT!

Minne

324: **1** (t heltal)

325: **5** (s heltal)

326: **10** (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

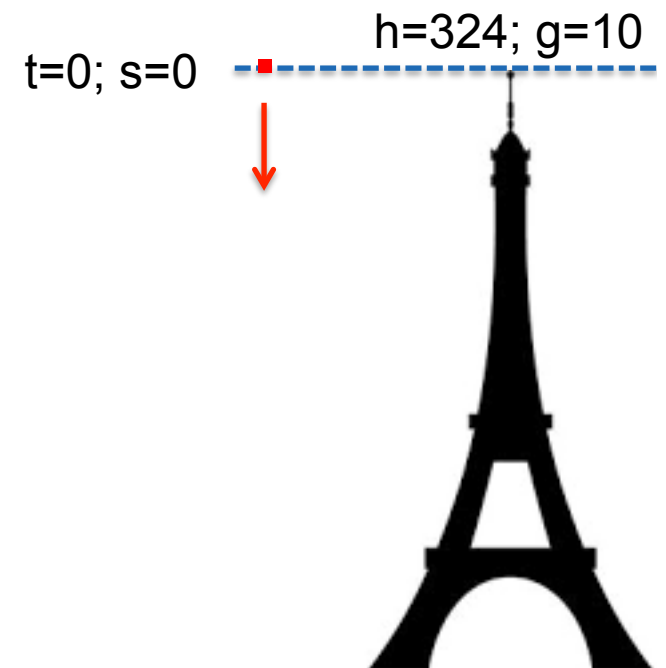
...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10



Minne

324: 1 (t heltal)

325: 5 (s heltal)

326: 10 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

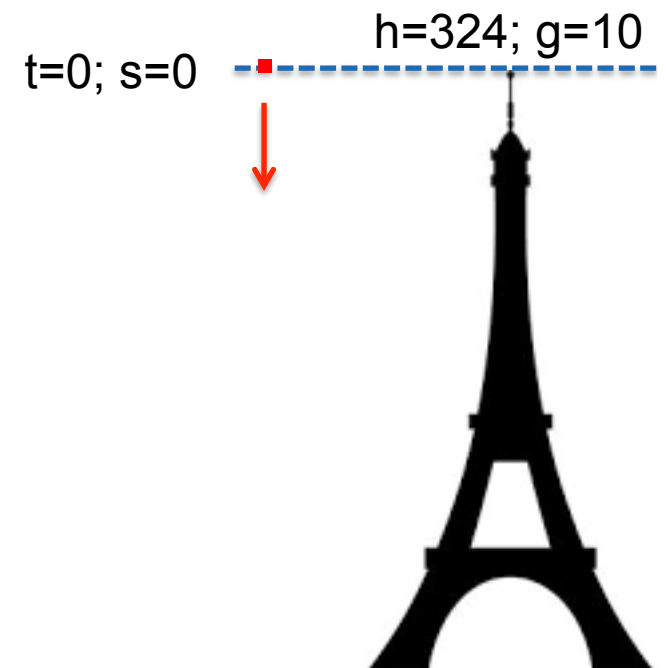
...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10



Minne

324: 2 (t heltal)

325: 5 (s heltal)

326: 10 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

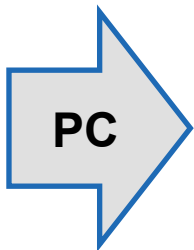
t = t + 1

v = 10*t

s = 0.5*10*t*t

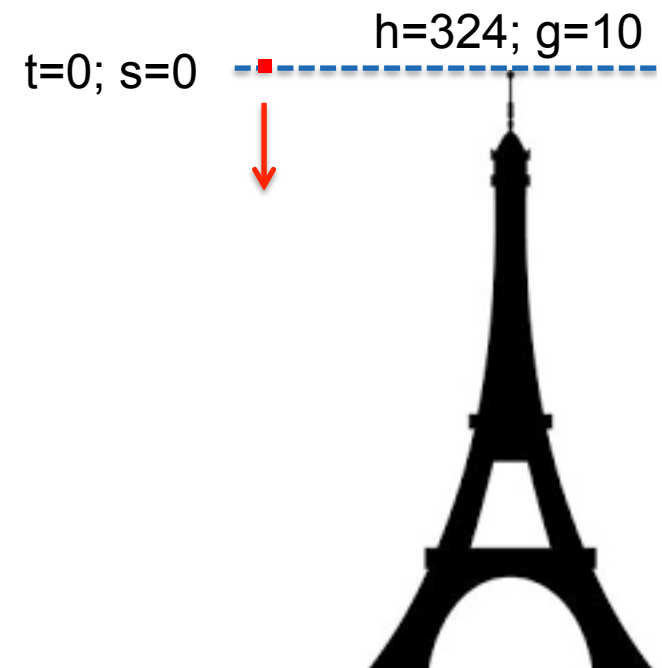
...så länge s<324

Visa klart!



Linjär acc!!!

t	v	s
0	0	0
1	5	10



Minne

324: 2 (t heltal)

325: 5 (s heltal)

326: 20 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

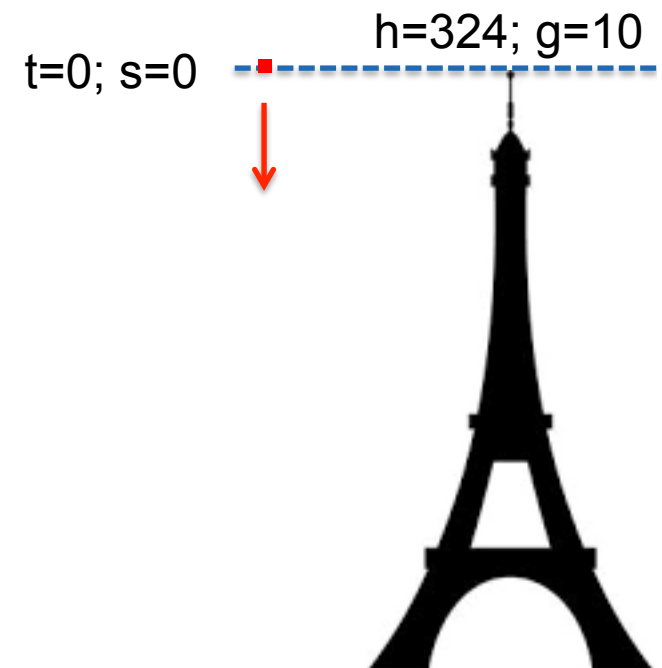
...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10



Minne

324: 2 (t heltal)

325: 20 (s heltal)

326: 20 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
---	---	---

0	0	0
---	---	---

1	5	10
---	---	----

SANT!

t=0; s=0

h=324; g=10



Minne

324: 2 (t heltal)

325: 20 (s heltal)

326: 20 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

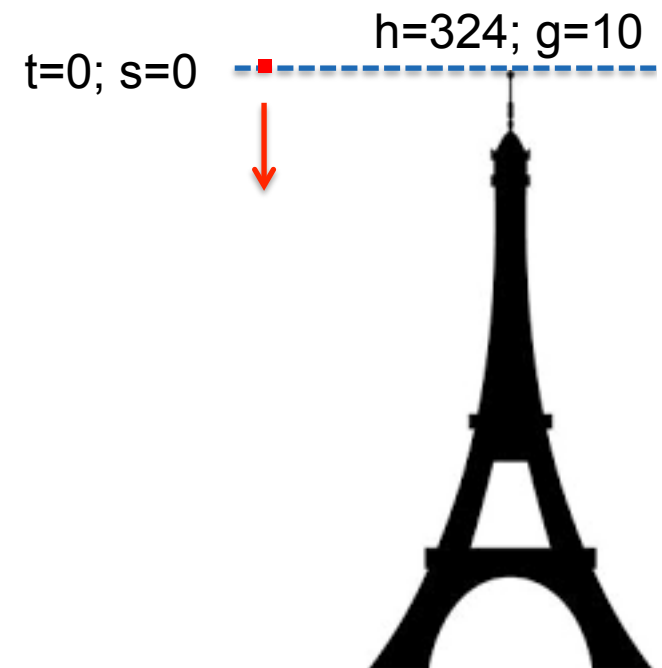
s = 0.5*10*t*t

...så länge s<324

Visa klart!

Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20



Minne

324: 2 (t heltal)

325: 20 (s heltal)

326: 20 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

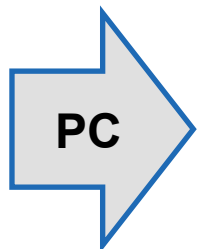
t = t + 1

v = 10*t

s = 0.5*10*t*t

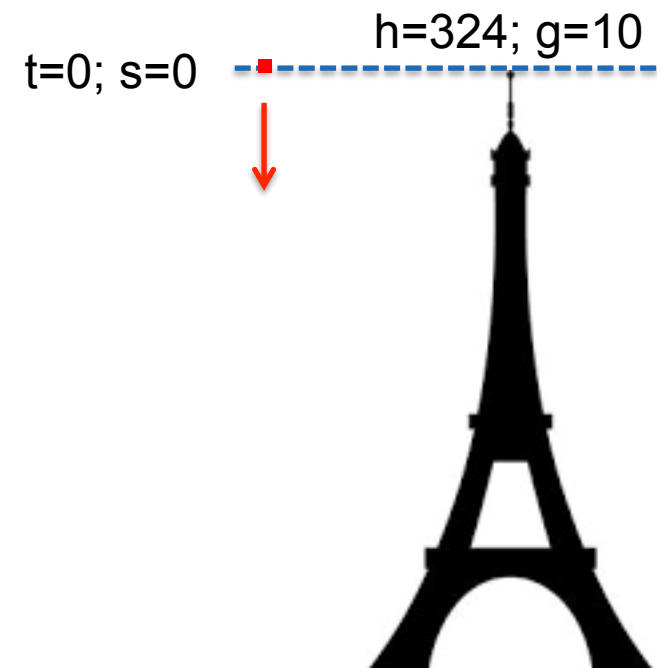
...så länge s<324

Visa klart!



Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20
*		
8	80	320



Minne

324: 8 (t heltal)

325: 320 (s heltal)

326: 80 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

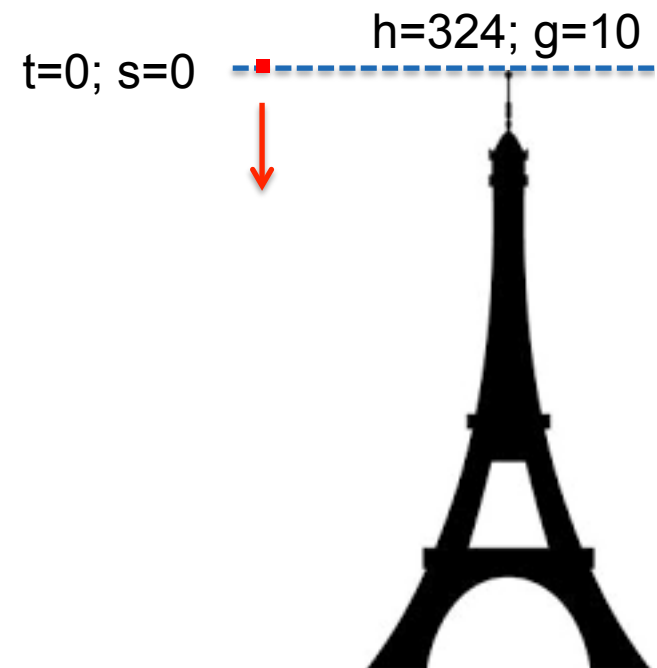
...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20
*		
8	80	320



Minne

324: 9 (t heltal)

325: 320 (s heltal)

326: 80 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

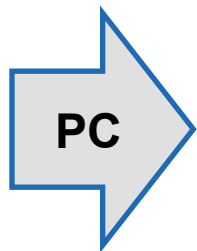
t = t + 1

v = 10*t

s = 0.5*10*t*t

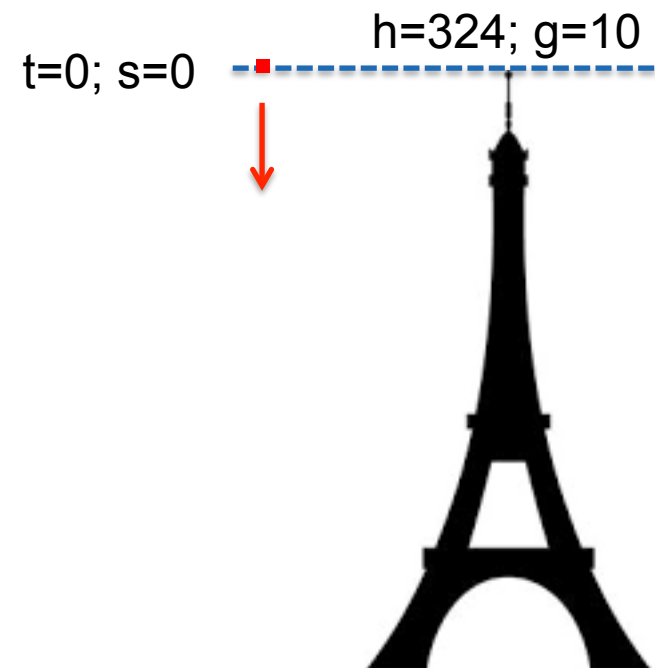
...så länge s<324

Visa klart!



Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20
*		
8	80	320



Minne

324: 9 (t heltal)

325: 320 (s heltal)

326: 90 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

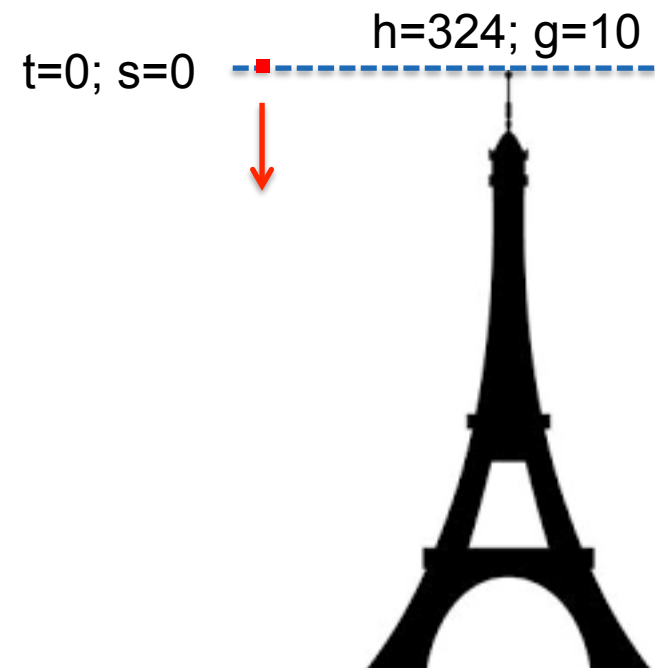
...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20
*		
8	80	320



Minne

324: 9 (t heltal)

325: 405(s heltal)

326: 90 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \&$$
$$s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

...så länge s<324

Visa klart!

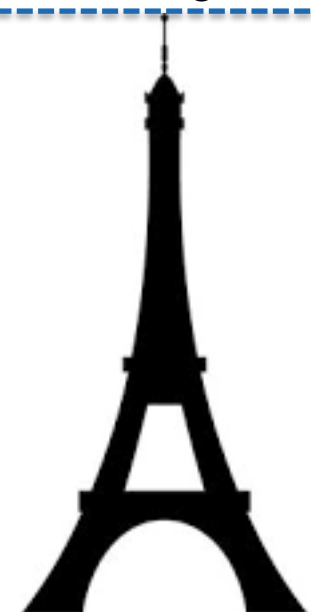
PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20
*		
8	80	320

t=0; s=0

h=324; g=10



FALSK!

Minne

324: 9 (t heltal)

325: 405 (s heltal)

326: 90 (v heltal)



Järnkoll?

Tabell för v & s för varje sekund tills bollen träffar marken!

$$v=at \quad \& \\ s=0.5at^2$$

t=0, s=0, v=0

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t, s & v

t = t + 1

v = 10*t

s = 0.5*10*t*t

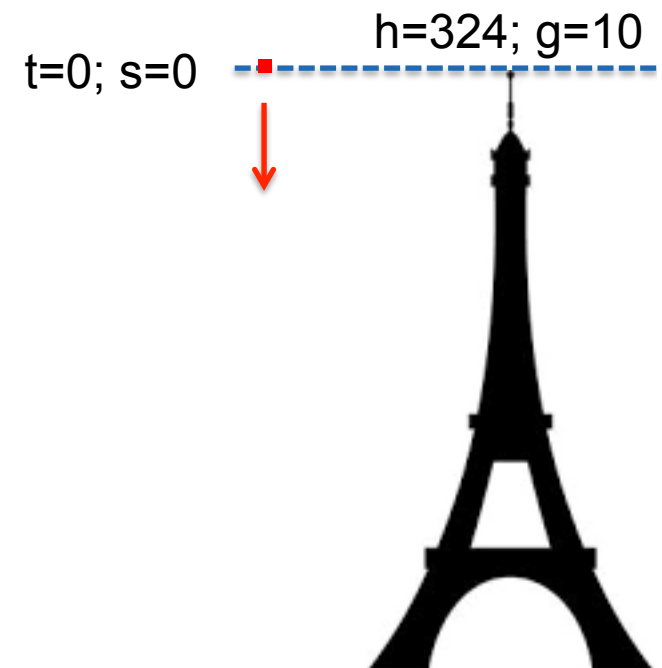
...så länge s<324

Visa klart!

PC

Linjär acc!!!

t	v	s
0	0	0
1	5	10
2	20	20
*		
8	80	320
Klart!		



Minne

324: 9 (t heltal)

325: 405(s heltal)

326: 90 (v heltal)



Järnkoll?

$t=0$, $s=0$, $v=0$

Visa programtitel

Visa tabellrubrik

Upprepa...

visa t , s & v

$t = t + 1$

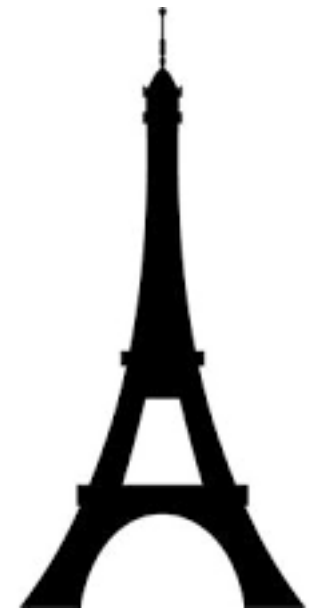
$v = 10 * t$

$s = 0.5 * 10 * t * t$

...så länge $s < 324$

Visa klart!

```
Constant acc demo 1.0 AC
time:speed:distance
0: 0: 0
1: 10: 5
2: 20: 20
3: 30: 45
4: 40: 80
5: 50: 125
6: 60: 180
7: 70: 245
8: 80: 320
Ground [8..9] sek!
Program completed, bye!
```





Introduktionskurs i Datateknik

F1: Vad är en dator?

F2: programmeringsspråk?

L1: Hour of code, alla kan programmera, Diplom!

L2: Visuellt programmering av svårare problem!



F3: Webbprogrammering – JavaScript!

L3: IDE, Web sida & JavaScript!



HTML





Kursmål

Sidor märkta med den här symbolen innehåller begrepp som är extra viktiga för dina fortsatta studier!



Vad är Internet?

Hypertext Markup Language (HTML).

Web server v.s. Web browser.

Hypertext Transfer Protocol (HTTP).

JavaScript.

Client/Server v.s. Peer-to-peer!



Vad är Internet?

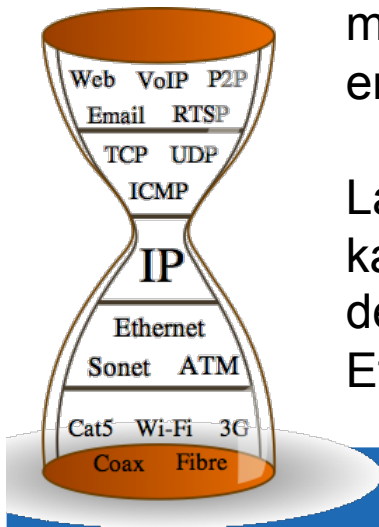


Termen internet är inte konstigare än att den beskriver flera sammankopplade datornät, nät av nät, internet!

Under 1970-talet började datornät bli vanligt förekommande men det var svårt att koppla ihop dem av tekniska skäl.

Robert Kahn och Vinton Cerf arbetade med problemet och skapade en lösning där det fanns ett nytt kommunikationslager, med unika adresser för alla datorer, som gjorde det relativt enkelt att knyta samman olika nätverk. Första näten -1982.

Lagret kallades Internet Protokoll (IP), och de unika adresserna kallades för IP-adresser. Om vi talar om det världsomspännande nät av ip-nät som vi t.ex. surfar via skriver vi Internet (stort I). Ett företagsinternt ip-nät kallas för ett intranet.



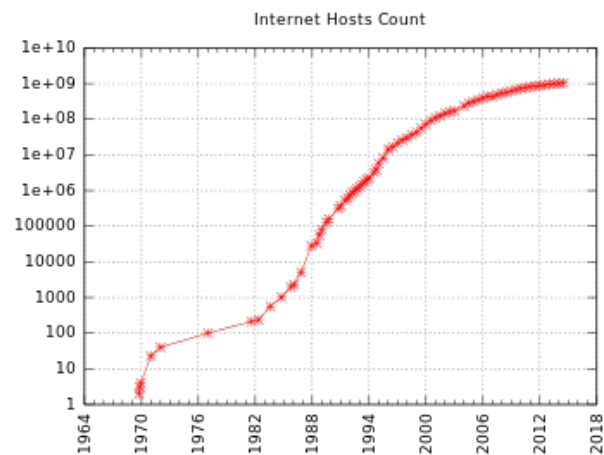
En ip-adress består av 4 tal (0..255) med punkter mellan talen.

130.237.84.8

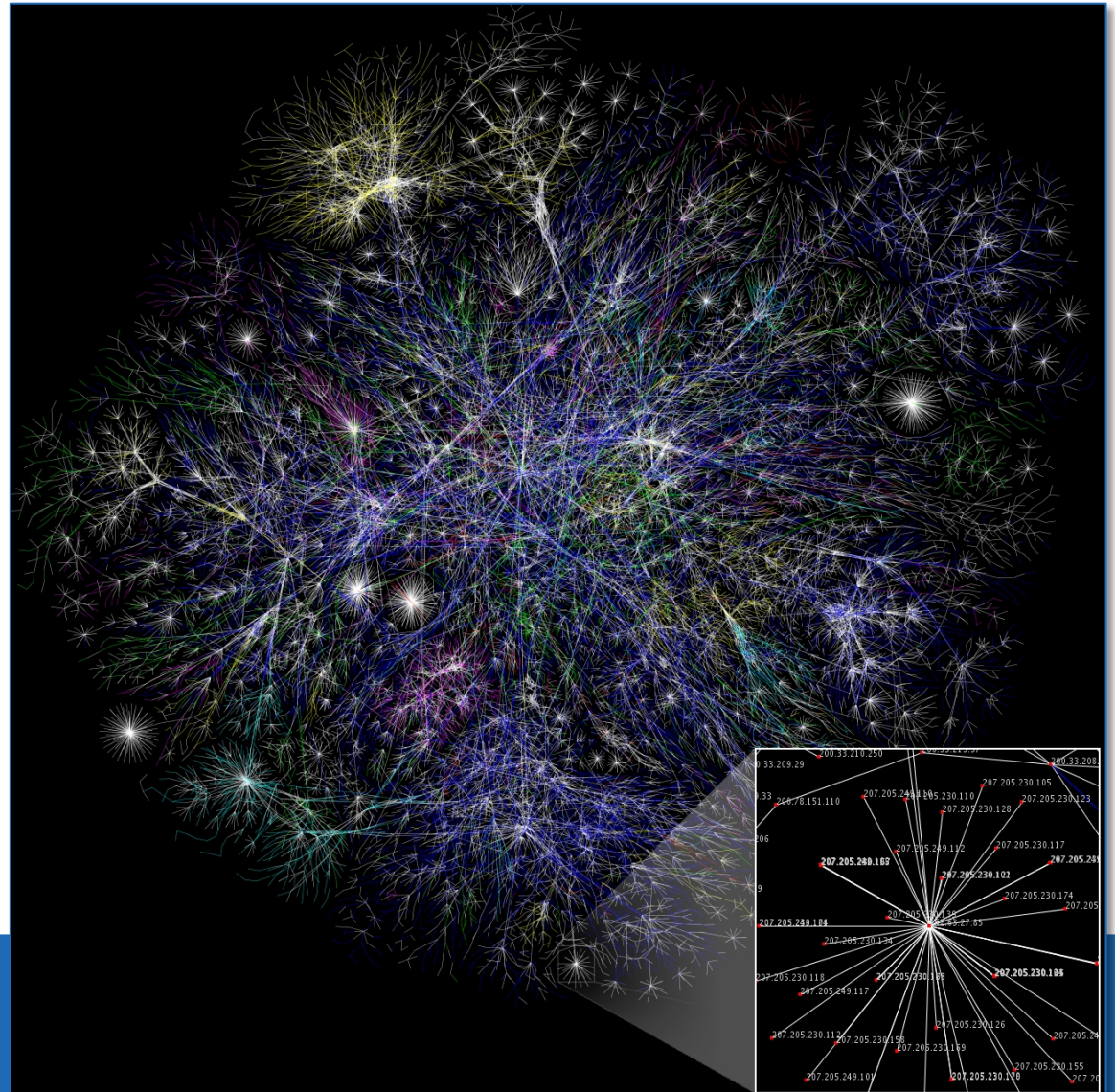


Internet

Via Internet går det i dag att nå över 1 miljard datorer men från början var det få. E-post och filöverföring var tidiga tjänster...



...men det var först när Berners-Lee uppfann WWW -91 som det tog riktig fart!





Vad var det Tim uppfann?!

URL

Protokoll

::/

Host

:

Port

/

Path

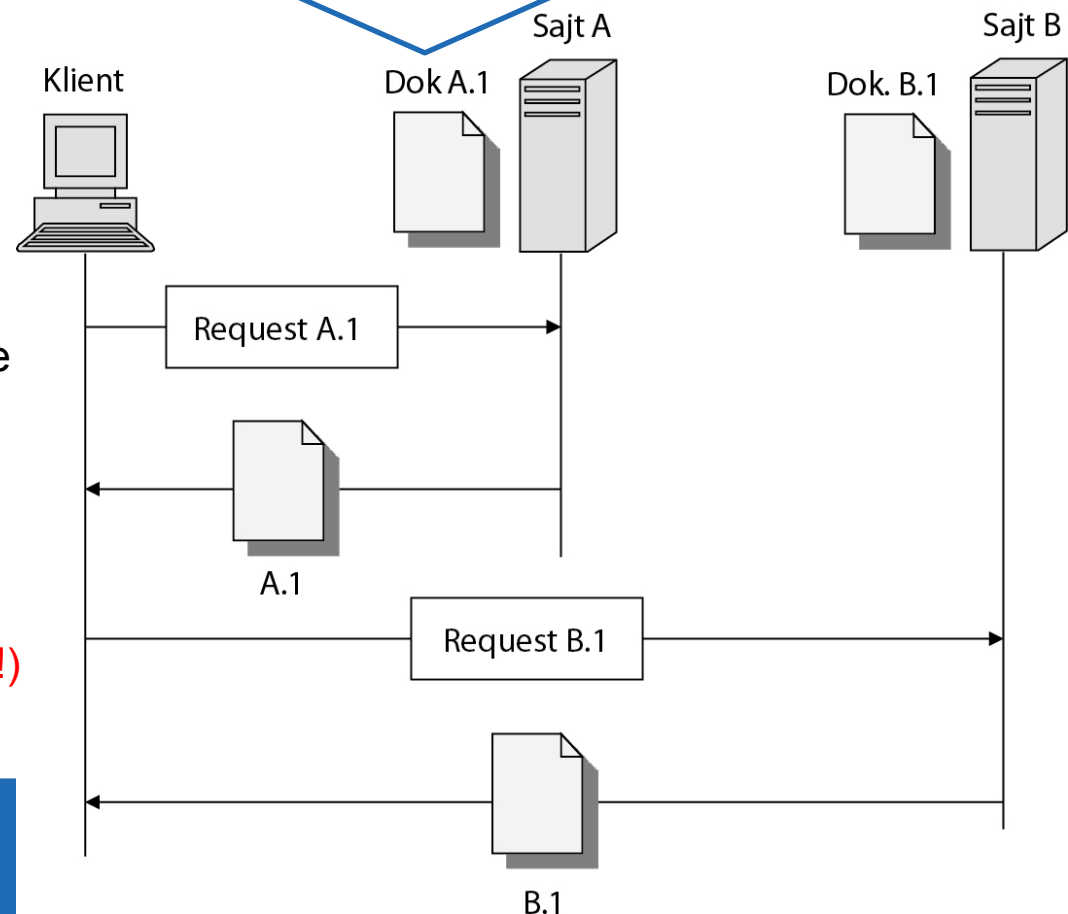
HTTP = Hypertext Transfer Protocol

Hur en klient (Web browser) hämtar en **specifik fil** från en Web server!!!

HTML = Hypertext Markup Language

Hur innehållet i den filen ska vara **uppmärkt** för att klienten ska kunna presentera innehållet!

(OBS: Inte ett programmeringsspråk!)





Att "märka upp" en text?!

A Sample Document

Rubrik

Wikipedians

June 23, 2007

Brödtext

This tiny sample showcases $\text{\TeX}$ and $\text{\LaTeX}$ for Wikipedia. More comprehensive examples, such as L^amport's `sample.tex` (see [1]) are available with any standard $\text{\TeX}$ distribution, or via www.tug.org. Only $\text{\TeX}$ nicians need [2].

1 Documents, Sections and Paragraphs

Kapitel

A $\text{\LaTeX}$ source file begins with a class declaration, followed by a preamble section, and the body of the document. The preamble contains settings and definitions. The body can contain sections, subsections, paragraphs, lists, equations, footnotes¹, figures and tables, and also commands for automatically-generated structures such as a table of contents, bibliography, index and glossary.

Commands such as `\section` introduce a new section, whilst an empty line delimits paragraphs. Other surplus white space is usually ignored.

2 Features

Both $\text{\TeX}$ and $\text{\LaTeX}$ allow for accents, and excel at typesetting mathematical equations, both in-line or displayed on a line by itself. For instance, an article on quadratics may need

$$ax^2 + bx + c = 0 \implies x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a},$$

Redan på -60-talet fanns det sätt att "märka upp" text så att ett tryckeri kunde trycka texten snyggt! Metoderna var olika, krångliga och främst för vetenskapliga texter...

...Tim skapade ett litet praktiskt mängd av "**taggar**" för vanlig text...

<h1>Rubrik nivå 1</h1>

<h2>Rubrik nivå 2</h2>

<i>Kursiv stil</i>

Det var sen upp till web-läsaren hur den skulle presentera innehållet! T.ex. kunde en tabell se helt olika ut i två olika läsare...



Nuvarande standard HTML5

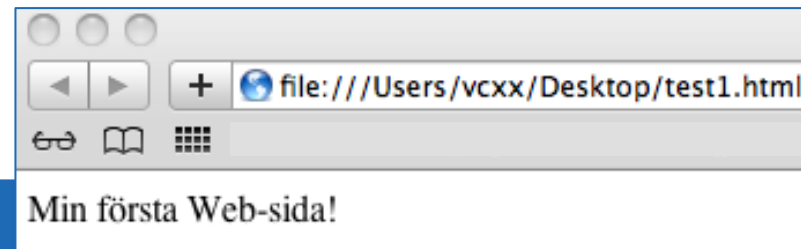


En minimal "uppmärkt" text ser ut så här!

Hela sidan! — {
Kontroll {
Innehåll {
—

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
    Min första Web-sida!
  </body>
</html>
```

Det här är en HTML5 uppmärkt sida!





Några till taggar...

`<h1>h1 tag</h1>`

`<h2>h2 tag</h2>`

`<h3>h3 tag</h3>`

`<h4>h4 tag</h4>`

`<h5>h5 tag</h5>`

`<h6>h6 tag</h6>`

`<p>Nytt stycke (paragraph)</p>`

"Extra"
mellanslag
& radslut
ignoreras!

UL – Unordered lists:

Example:

- Apples
- Oranges
- Bananas

UL Code:

```
<ul>
  <li>List item 1</li>
  <li>List item 2</li>
</ul>
```

OL – Ordered Lists:

Example:

1. Breakfast
2. Lunch
3. Dinner

OL Code:

```
<ol>
  <li>List item 1</li>
  <li>List item 2</li>
</ol>
```

h1

h2

ul

h2

ol

PIC Microcontrollers

Identification

- ISBN-13:978-0-13-119404-5
- ISBN-10:0-13-119404-6

Chapters

1. Introduction to Computing
2. The PIC Microcontrollers
3. Architecture
4. Branch, Call & Skip

LAB #3!
....TEST....



CSS & JavaScript

Dagens Web-sidor kräver att utvecklaren är insatt i tre områden!

- HTML – Som märker upp och specificerar innehållet på sidan.
- CSS – Som specificerar sidans utseende (layout).
- JavaScript – Som skapar sidans interaktiva egenskaper.

Två korta exempel...



CSS

- Före HTML5 fanns det taggar/attribut som indikerade hur den som skapat sidan ville att den skulle visas, men de var inte standardiserade eller heltäckande!
- Med CSS går det på pixelnivå att styra allt som visas!
- CSS-informationen ligger vanligen i en separat fil...
- ...men den kan skrivas i samma fil som innehållet.

PIC Microcontrollers

Identification

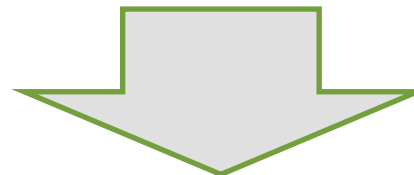
- ISBN-13:978-0-13-119404-5
- ISBN-10:0-13-119404-6

Chapters

1. Introduction to Computing
2. The PIC Microcontrollers
3. Architecture
4. Branch, Call & Skip



`<ul style="font-weight: bold">`



PIC Microcontrollers

Identification

- ISBN-13:978-0-13-119404-5
- ISBN-10:0-13-119404-6

Chapters

1. Introduction to Computing
2. The PIC Microcontrollers
3. Architecture
4. Branch, Call & Skip



JavaScript

- De första webbsidorna var helt statiska men snart skapades ett enkelt programmeringsspråk, JavaScript, som gjorde det möjligt att skapa webbsidor som interagerade med läsaren!
- JavaScript (som inte ska blandas ihop Java) kan:
 - Ändra innehållet på en sida,
 - Ändra attribut på en sida,
 - Ändra CSS på en sida,
 - Kontrollera att formulär är korrekt ifyllda!

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <script>
      function show() {
        alert(Date());
      }
    </script>
  </head>
  <body onClick="show()">
```





Webbutveckling...

- Vi ska snart tala om Programutveckling men du kommer säker att brottas med Webbutveckling i någon form när du börjar arbeta.
- Webbutveckling sker i dag alltid med hjälp av någon WYSIWYG-utvecklingsmiljö men som du har förstått är det bra att förstå hur man "märker upp" innehåll, lite känsla för layout samt viss insikt i programmering för interaktivitet.
- Den som är intresserad bör ta del av W3Schools utmärkta webbplats där det går att lära sig allt om HTML, CSS & JavaScript!

w3schools.com

THE WORLD'S LARGEST WEB DEVELOPER SITE

TUTORIALS ▾

REFERENCES ▾

EXAMPLES ▾





WIKI

Även om det är lätt att skapa en webbsida, vilket i dag alltid görs i någon WYSIWYG-editor, så krävs en viss kunskap för att det ska bli bra.

-04 myntades begreppet Web 2.0 för en ny typ av webbinnehåll där fokuset låg på användargenererat innehåll och användbarhet. Sociala medier tog fart och världen fick Bloggar, Tweets och Wikis.

En Wiki är ett otroligt bra verktyg för all form av dokumentation, individuellt eller i grupp!!!





4 853 000+ articles

1 806 000+ Artikel

962 000+ 記事

814 000+ 條目

1 106 000+ hasat



1 172 000+ artículos

1 213 000+ статей

1 614 000+ articles

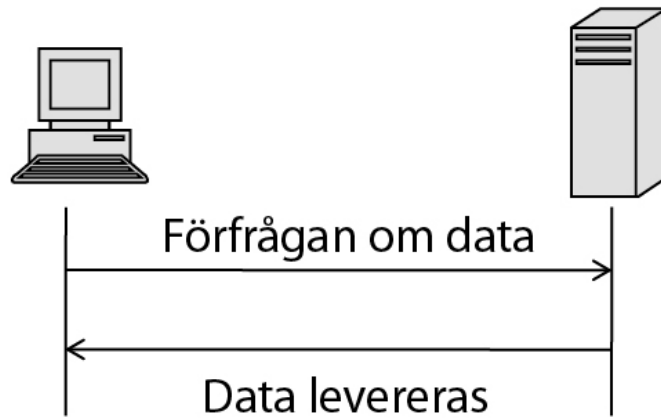
1 193 000+ voci

871 000+ artigos

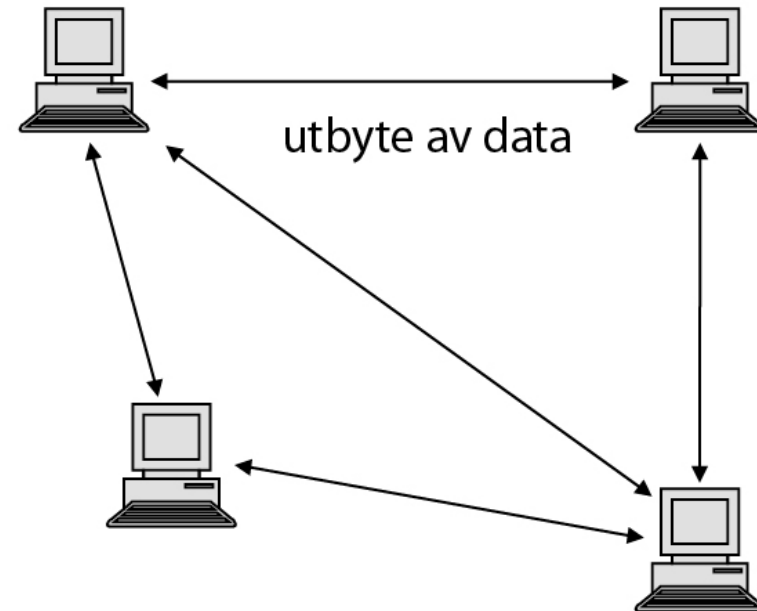
Tips: Använd en Wiki direkt i alla projekt!

Client/Server v.s Peer-to-peer...

(a)



(b)





Programmering...

1. Vi såg i föreläsning #1 att t.o.m dagens datorer inte förstår något annat språk än sitt eget maskinspråk...
2. I föreläsning #2 tittade vi på programmeringsspråk. De var lättare att förstå men begränsade (på olika sätt). För att programmera krävs det en ordbehandlare och en kompilator!
3. Det går att skriva "programmet" i nästan vilken ordbehandlare som helst, och sedan kompilera programmet men i dag använder alla en integrerad utvecklingsmiljö som underlättar alla steg i processen...
4. I kursen Grundkurs i programmering ska ni använda utvecklingsmiljön CodeLite som introduceras här, och som ni kommer att få installera i lab #3!

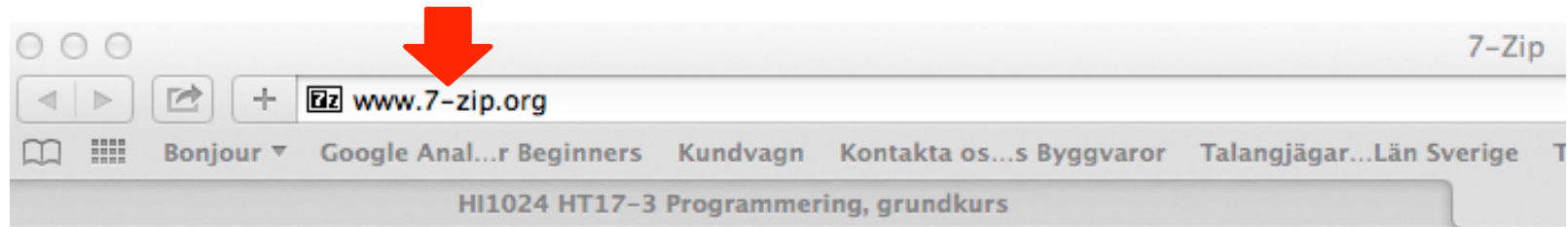


**A Powerful IDE for
C/C++
Supporting all major
compilers**



Installera CodeLite...

STEG 0: ENBART FÖR PC!



7-Zip is a file archiver with a high compression ratio.

Download 7-Zip 16.04 (2016-10-04) for Windows:

Link	Type	Windows	Size
Download	.exe	32-bit x86	1 MB
Download	.exe	64-bit x64	1 MB

7-Zip

License

Enbart för pc!

HÄMTA OCH INSTALLERA "PACKA UPP" PROGRAMMET 7-ZIP!



Installera CodeLite...

!!!Virusskydd!!!
MAC: Systeminställningar/Säkerhet & Integritet

STEG 1: Hämta CodeLite!

CodeLite IDE Products Download Screenshots GitHub Wiki Forums Blog wxCrafter [gitter](#) [join chat](#)

A Free, open source, cross platform C,C++,PHP and Node.js IDE

CodeLite 10.0 - Stable Release released on Jan 05, 2017 [What's New?](#)

CodeLite 10.0 for Windows 64 bit Installer Direct Link GitHub	sha1sum: 378ed383631a25e64df9ec5f7e2709195fcd30d3
CodeLite 10.0 for Windows 32 bit Installer Direct Link GitHub	sha1sum: d83fc6b0c6f185d0407e6dbdf11098dcb7825079
CodeLite 10.0 App Bundle for OSX 10.10 Direct Link GitHub	sha1sum: de2c94045971f8e0a9f3219ccc47a2d83cc22ce5

För Mac & pc!

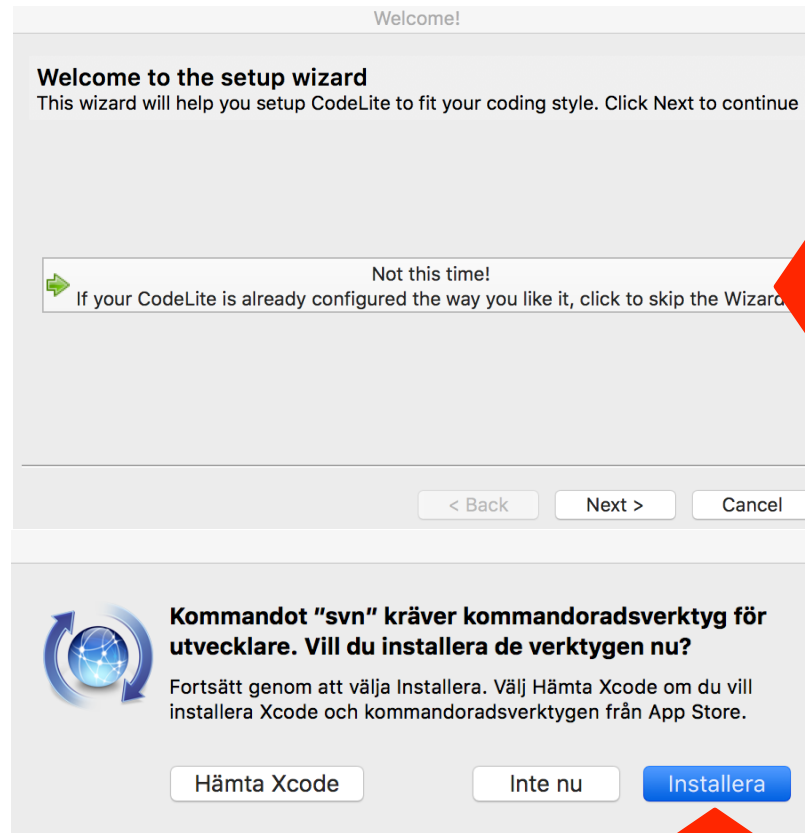
HÄMTA CodeLite...



Installera CodeLite...

Enbart Mac...

STEG 2: Installera CodeLite!



...välj skip...

...välj installera - klart!

För Mac

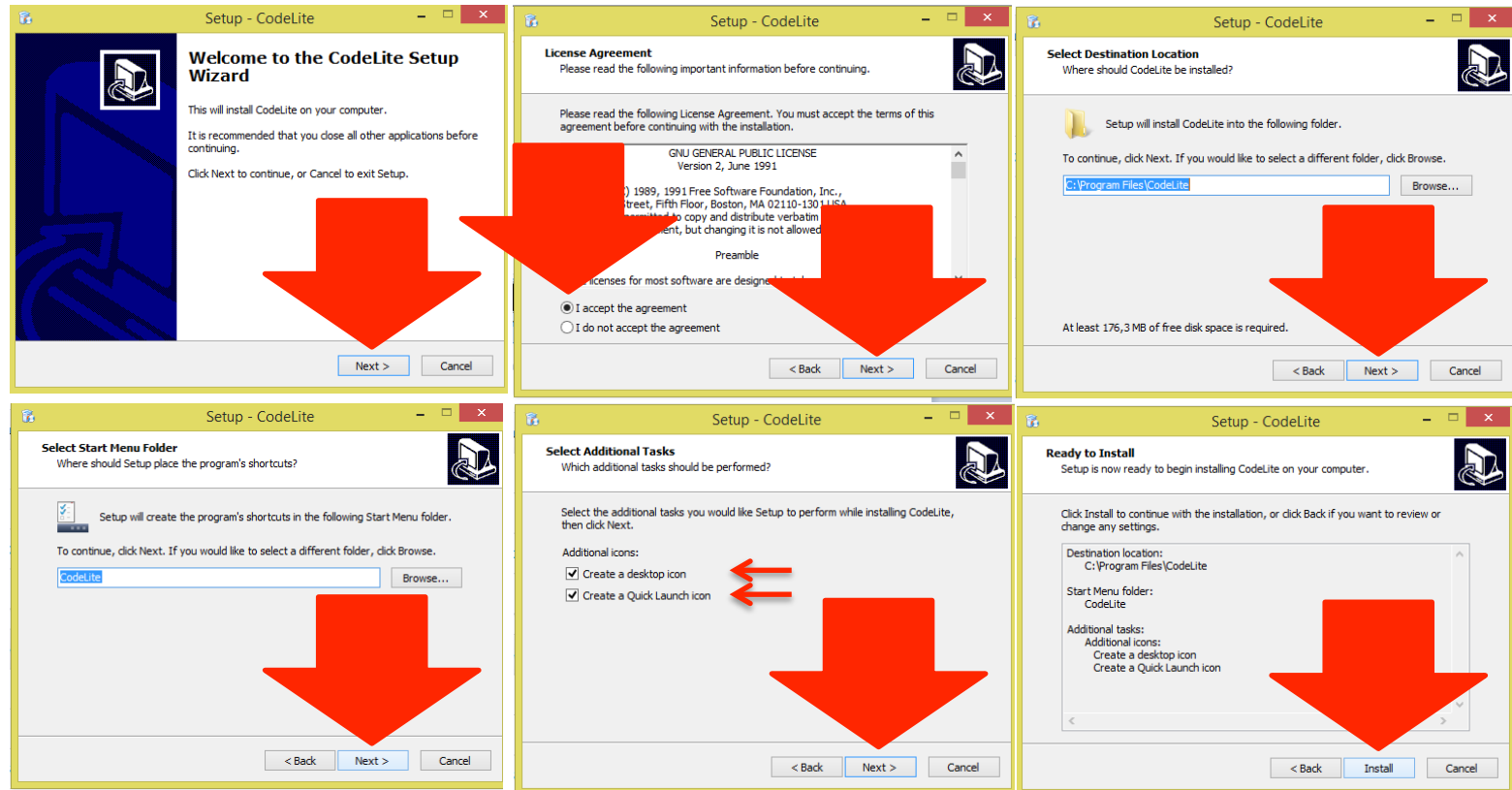
...Installera CodeLite, 2 steg!



Installera CodeLite...

Enbart PC...

STEG 2: Installera CodeLite!



(nu börjar den installera...)

För PC

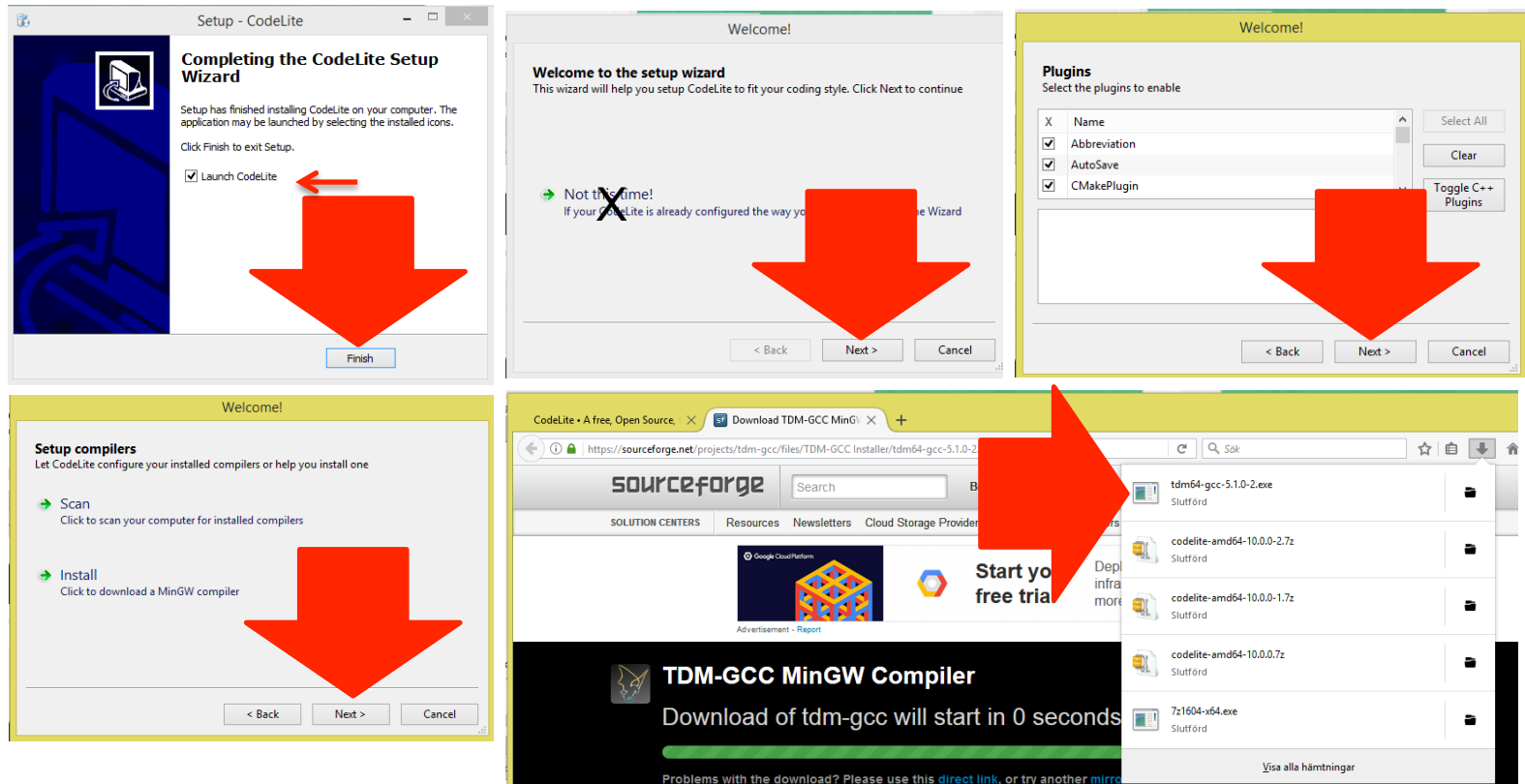
...Installera CodeLite, de första 6 stegen...



Installera CodeLite...

STEIG 2: Installera CodeLite!

Enbart PC...



(Så snart nerladdningen av MinGW kompilatorn är klar, starta installationen av den (om det inte sker automatiskt)!)

För PC

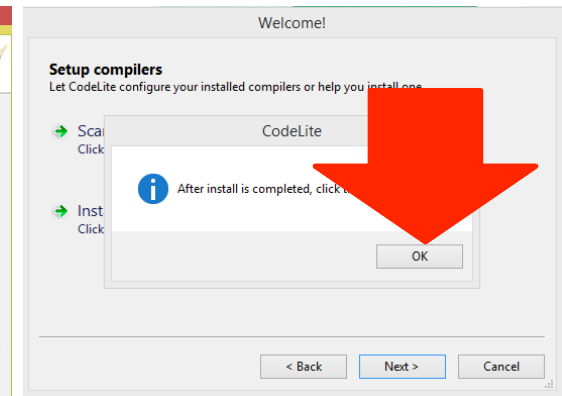
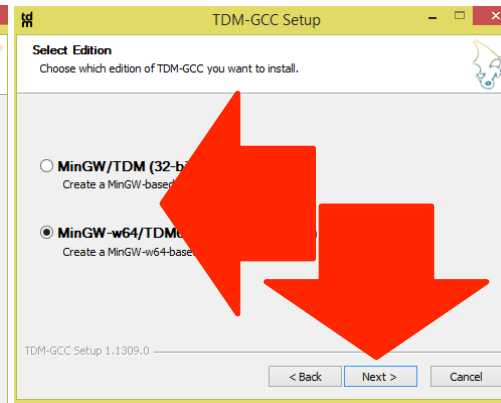
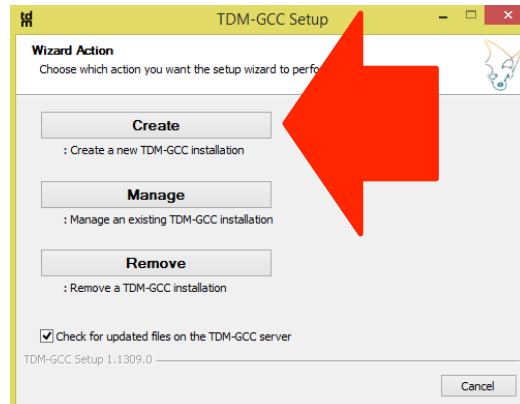
...Installera CodeLite, 5 stegen till...



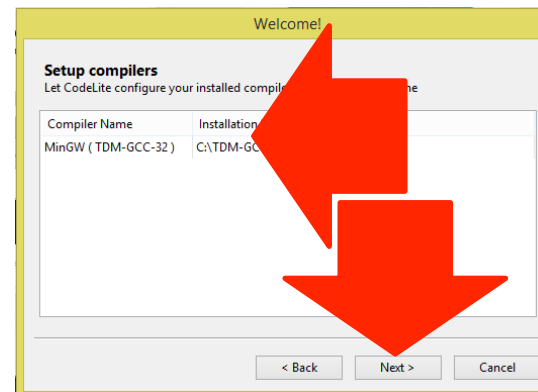
Installera CodeLite...

Enbart PC...

STEG 2: Installera CodeLite!



Tryck på Scan!



Next, Next Next...

Klar!

För PC

...Installera CodeLite, de sista X stegen!



STEIG 3: Sökväg kompilator!

Installera CodeLite...

VÄNTA MED DETTA!
GÄLLER INTE ALLA
VER. AV WINDOWS

Enbart PC...

Högerklicka på **computer** (den här datorn?) och välj **properties**.

Välj **advanced system and settings**.

Välj **Environment variables** (miljövariabler).

Välj **path**-variablen och välj **edit**.

Var noga så att du inte tar bort något som redan finns.

Vi ska bara lägga till.

Lägg till ett ; sist i raden och klipp sedan in sökvägen till gcc:.

I min installation:

;C:\TDM-GCC-64\bin

Öppna nu kommandotolken (cmd) och prova att skriva gcc.

Om den klagar över att den saknar input-fil har vi lyckats:

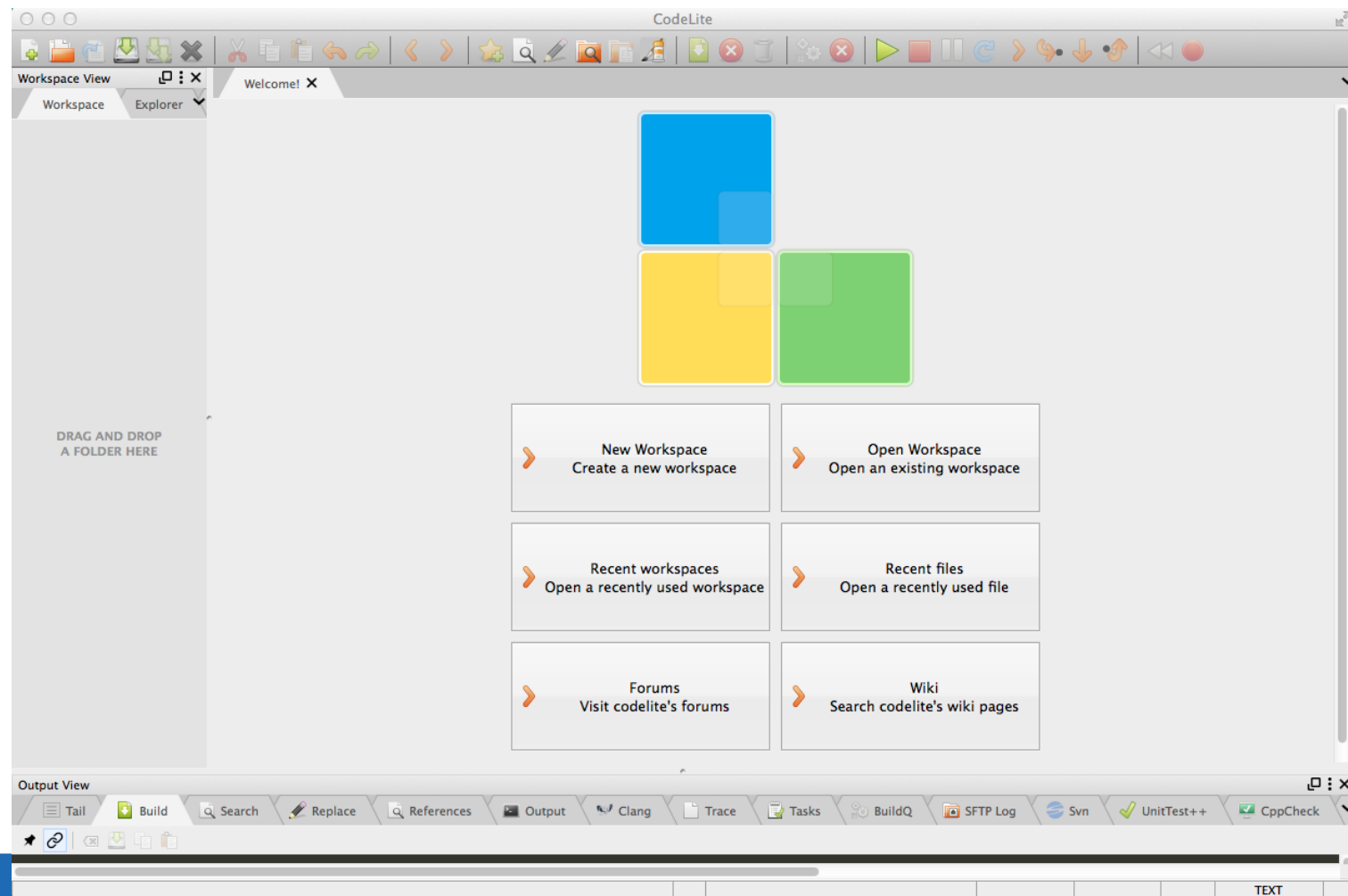
gcc: fatal error: no input files compilation terminated.

För PC

...gör detta steg om du inte kan kompilera, nästa exempel!



Använda CodeLite...



Starta CodeLite... ...det borde se ut ungefär som på bilden!



Använda CodeLite...

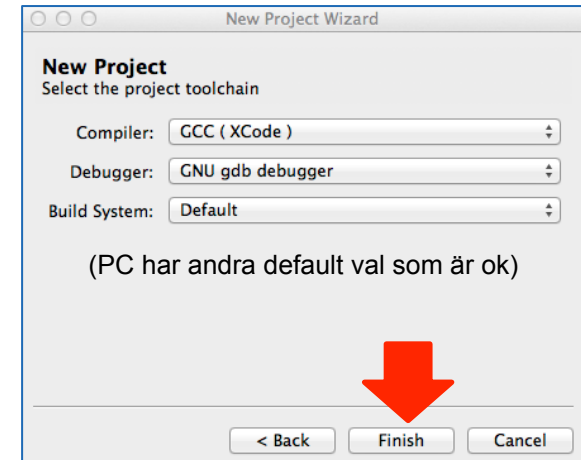
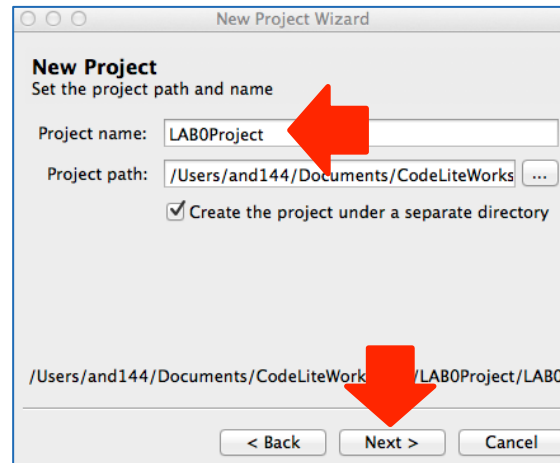
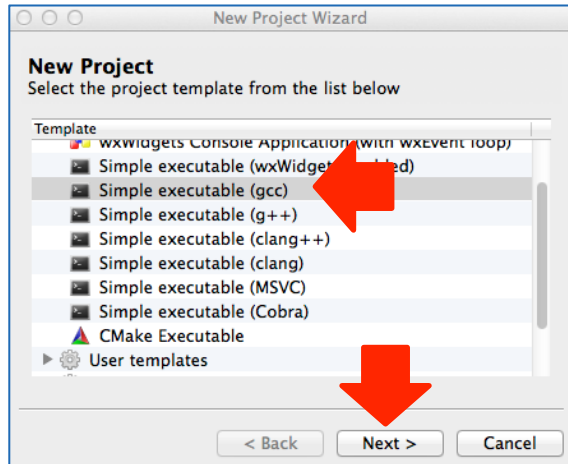
The screenshot shows the CodeLite IDE interface. On the left, the 'Workspace View' panel is visible with a red arrow pointing to the '(TOMT)' (Empty) workspace. In the center, a large red arrow labeled '1' points to the 'New Workspace' button in the 'Welcome!' dialog. To the right, the 'New Workspace' dialog is open, showing the 'Workspace Name' field set to 'CodeLiteWorkspace' (indicated by red arrow 4) and the 'Workspace Path' set to '/Users/and144/Documents'. The 'Generated File' field shows the path '/Users/and144/Documents/CodeLiteWorkspace/CodeLiteW'. The checkbox 'Create the workspace under a source directory' is checked (indicated by red arrow 5). The 'OK' button is highlighted. Above the 'New Workspace' dialog, a 'Select the workspace...' dialog is shown with 'C++', 'Node.js', and 'PHP' listed. A red arrow labeled '2' points to the 'C++' option, and a red arrow labeled '3' points to the 'OK' button. At the bottom right, a green arrow points to the 'CodeLiteWorkspace' entry in the 'Workspace View' panel.

CodeLite behöver ett ställe att lagra dina filer...
...tryck på "create a new workspace"!

OK!

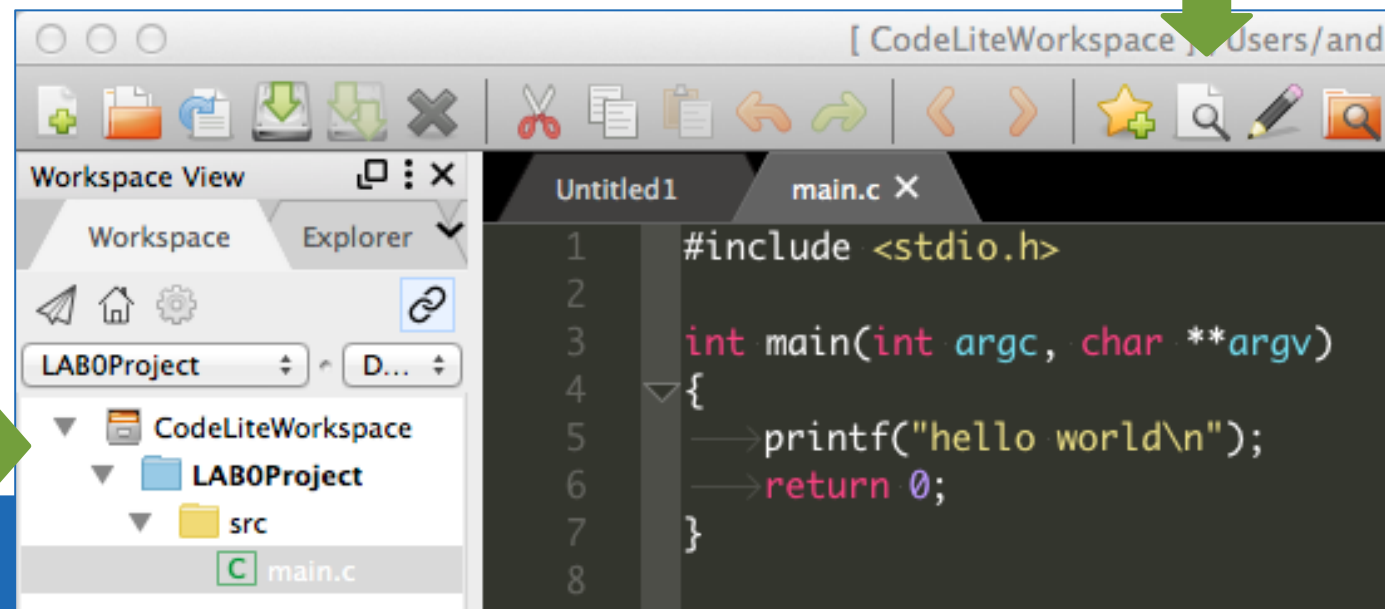


Använda CodeLite...



Välj nu:
File/New/
New project...

...som skapar
ett projekt med
en källkodsfil!





Använda CodeLite

Välj nu från **Build** menyn...

...**Build Project**, och därefter

...**Run**, då ska ett nytt fönster öppnas med bl.a. texten **Hello World!**



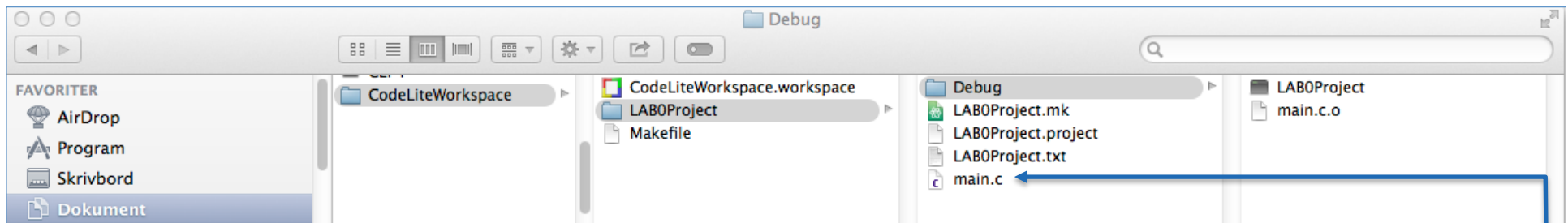
```
Debug — bash — 54x15
Last login: Mon Jul 24 17:24:33 on ttys009
anderss-mini:~ and144$ cd /Users/and144/Documents/Code
LiteWorkspace/LAB0Project/Debug && ./LAB0Project
hello world
anderss-mini:Debug and144$
```



Använda CodeLite (CL)

ÖVERKURS

Som ni kan se så skapar CL en avancerad filstruktur för varje projekt!



Mitt workspace... ..med 1 folder/projekt... ..folder för byggresultat... ..körbar fil m.m
& projektfiler (LAB0Project)
& källkoden main.c

Det finns inget som hindrar,
och det kan vara bra att
kunna, att kompilera i
kommandotolken...
(Tips: F2 om OS)

```
anderss-mini:Debug and144$ ls
LAB0Project      main.c.o
anderss-mini:Debug and144$ cd ..
anderss-mini:LAB0Project and144$ ls
Debug            LAB0Project.project      main.c
LAB0Project.mk   LAB0Project.txt
anderss-mini:LAB0Project and144$ gcc main.c -o main
anderss-mini:LAB0Project and144$ ls
Debug            LAB0Project.project      main
LAB0Project.mk   LAB0Project.txt          main.c
anderss-mini:LAB0Project and144$ ./main
hello  world
anderss-mini:LAB0Project and144$
```



Lab #3: Hello World!

Du ska installera CodeLite IDE, och utveckla ett lite klassiskt c-program!

("Hello World" från föregående sida!)



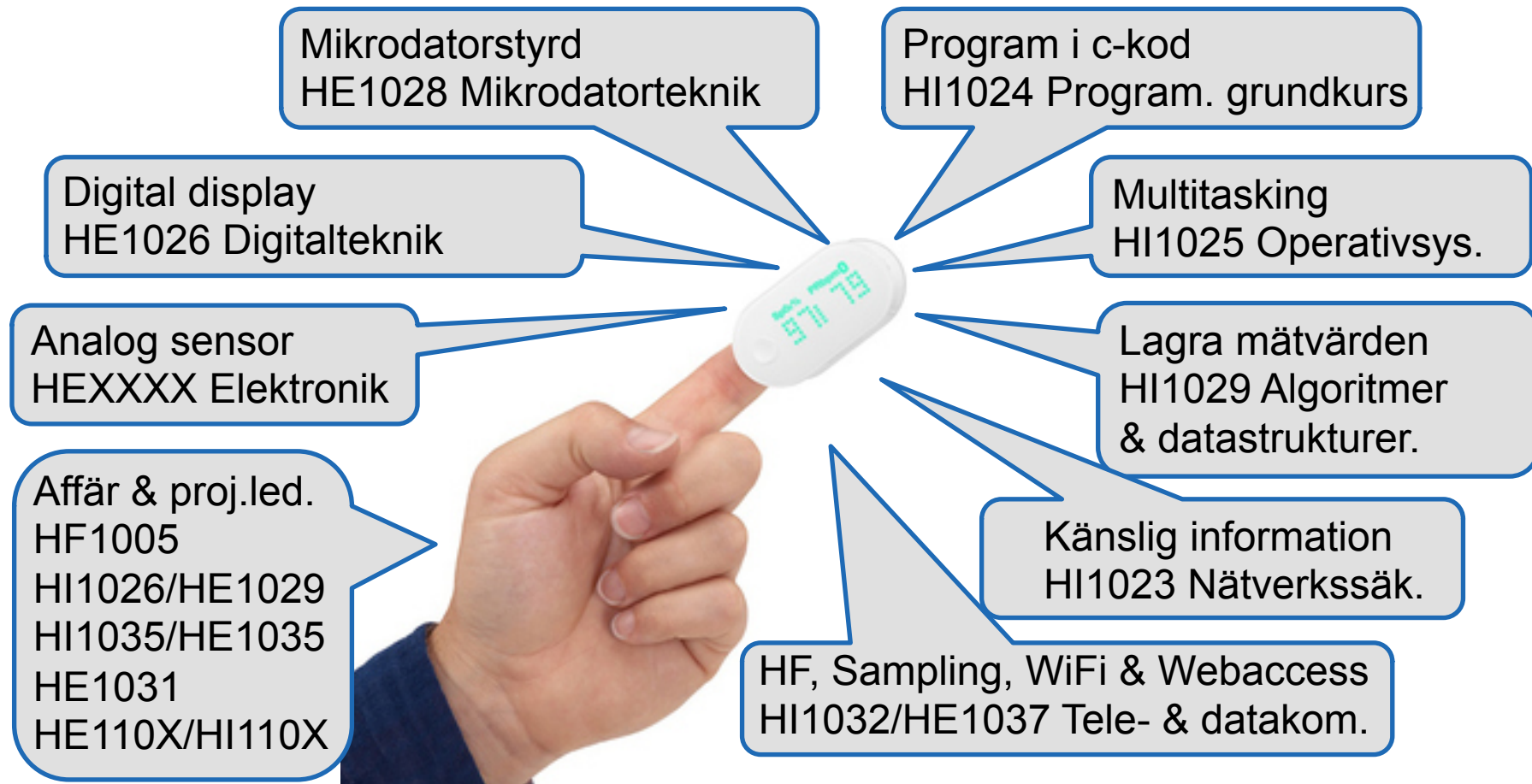
**A Powerful IDE for
C/C++
Supporting all major
compilers**

Grattis!

Du och din dator är nu klara för grundkurs programmering HI1024!



Din Innovation!



Exempel: The pulsmätare!



Rev history

150507 0.1 AC	Skapad.
150518 0.3 AC	Internet-delen nästan klar.
150522 0.5 AC	Första utkastet där allt är med på något sätt.
150823 1.0 AC	Expanderat Wiki-delen med stycke om att uppträda som ingenjör.
160615 1.1 AC	Inkluderat programutvecklingsmetodik som flyttats från föreläsning #2.
160704 2.0 AC	Uppsnyggat för HT16.
170709 3.0 AC	Uppdaterad för ny IDE Codelite.