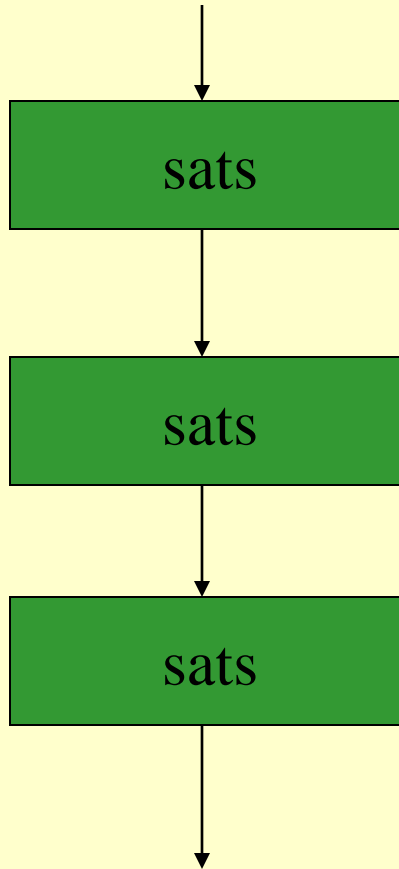


Rubrik

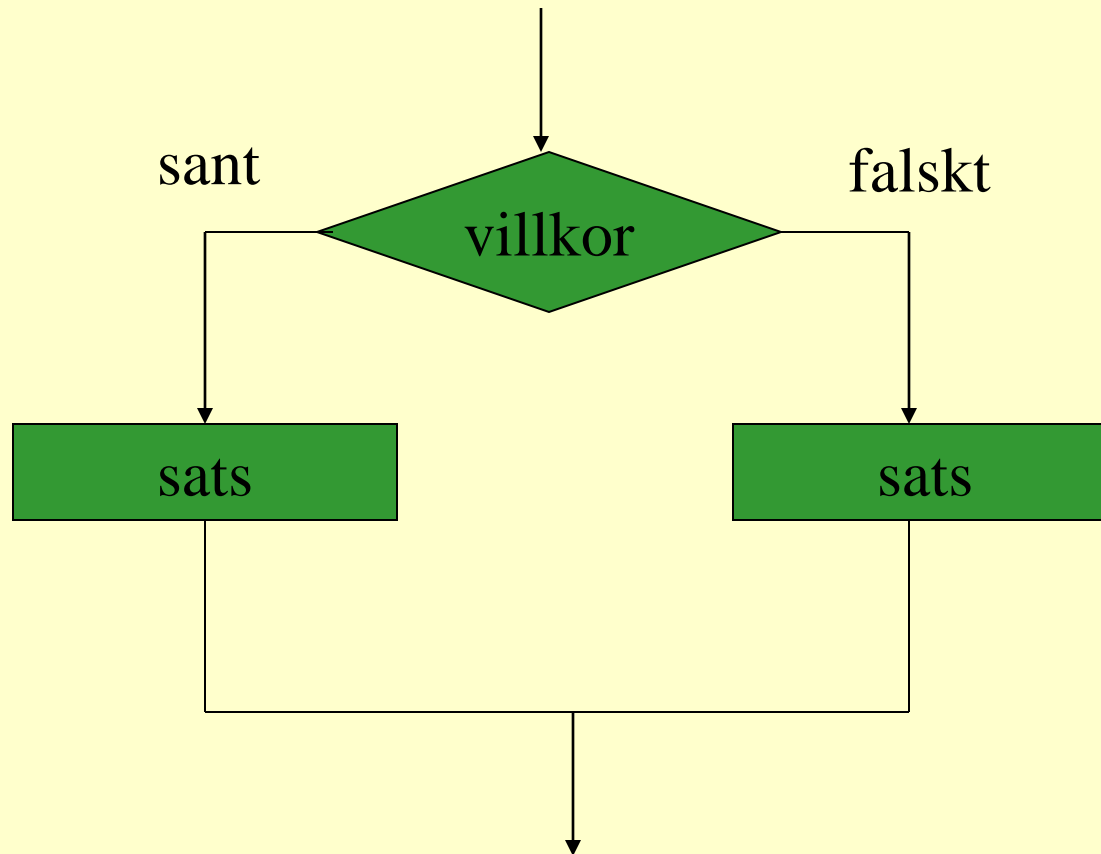
- For-loop
- While-loop
- printf, scanf
- slumpstal
- math.h

Påminnelse: Vi går inte igenom allt på föreläsningen – du måste läsa igenom boken och särskilt det vi inte hunnit med!

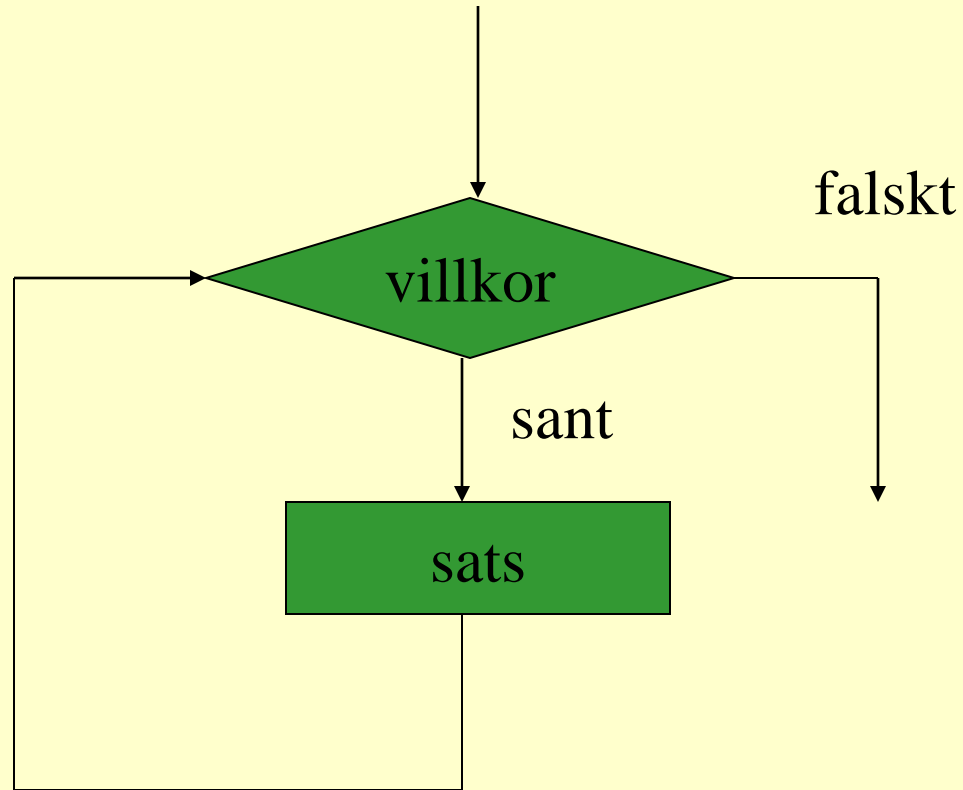
Sekvens



Selektion (if)



Iteration (for, while)



for

- for – emedan, så länge
- används främst när vi vet hur många gånger vi vill upprepa något

Exempel på for-slinga (loop)

Startsats, utförs först och endast en gång

Utförs sedan och först i varje påföljande varv. Om resultatet blir true utförs {}.

```
int n;  
for (n = 0; n < 10; n++)  
{  
    printf("%d, ", n);  
}
```

Utföres sist efter {} i varje varv. Används normalt för att stega upp eller ner n. ++ - betyder öka n med 1, dvs $n = n + 1$.

Resultatet blir: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9,

Fler eksempel

```
int n;  
for(int n=10;n>0;n--)  
{  
    printf("%d, ", n);  
}  
  
for(;1;)  
{  
    printf("evig loop\n");  
}
```

(ctrl-c i kommandotolken)

```
int k;  
int summa = 0;  
for(k=1;k<=10;k=k+1)  
{  
    summa = summa + k;  
    printf("Summa hittills: %d\n", summa);  
}
```

En körning ser ut så här:

Summa hittills: 1

Summa hittills: 3

Summa hittills: 6

Summa hittills: 10

Summa hittills: 15

Summa hittills: 21

Summa hittills: 28

Summa hittills: 36

Summa hittills: 45

Summa hittills: 55

Nästlade loopar

```
int i,j;
for(i=1;i<=7;i=i+1)
{
    for(j=1;j<=8;j=j+1)
    {
        printf("(%d,%d) ", i, j);
    }
    printf("\n");
}
```

Resultat:

(1,1) (1,2) (1,3) (1,4) (1,5) (1,6) (1,7) (1,8)
(2,1) (2,2) (2,3) (2,4) (2,5) (2,6) (2,7) (2,8)
(3,1) (3,2) (3,3) (3,4) (3,5) (3,6) (3,7) (3,8)
(4,1) (4,2) (4,3) (4,4) (4,5) (4,6) (4,7) (4,8)
(5,1) (5,2) (5,3) (5,4) (5,5) (5,6) (5,7) (5,8)
(6,1) (6,2) (6,3) (6,4) (6,5) (6,6) (6,7) (6,8)
(7,1) (7,2) (7,3) (7,4) (7,5) (7,6) (7,7) (7,8)

Max/Min av heltal

```
#include <stdio.h>

int main(void){
    int max,min,k,tal;
    printf("Tal nr 1: ");
    scanf("%d",&tal);
    min=tal;
    max=tal;
    for(k=2;k<=5;k++){
        printf("Tal nr %d: ",k);
        scanf("%d",&tal);
        if (tal<min) min=tal;
        if (tal>max) max=tal;
    }
    printf("%d %d\n",max,min);
}
```

Resultat:

Tal nr 1: 100

Tal nr 2: 50

Tal nr 3: -40

Tal nr 4: 2

Tal nr 5: 40

100 -40

while

- **while** betyder **medans** eller **så länge**

- `while(villkor)`

{

sats1

sats2

...

}

-satserna upprepas så länge villkor är sant

-Används när antalet upprepningar är okänt

Exempel

- ```
while(1){
 printf("det roliga tar aldrig slut\n");
}
```
- ```
int i = 0;  
while(i<10){  
    printf("%d, ", i);  
    i=i+1;  
}
```

```
#include <stdio.h>
int main(void)
{
    int n=1;
    while(n != 0)
    {
        scanf("%d",&n);
        printf("Du skrev:%d \n", n);
    }
}
```

do - while

- do
 {
 sats1
 sats2
 ...
 } while(*villkor*);

jämförelsen görs sist

körs minst en gång

ibland naturligare, tex i exemplet nyss

```
#include <stdio.h>
int main(void)
{
    int n;
    do
    {
        scanf("%d",&n);
        printf("Du skrev:%d \n", n);
    } while(n != 0);
}
```

printf

```
int h=7, k=14;
```

```
printf("Totalt %d hästar och %d kor", h, k);
```

Utskriften: Totalt 7 hästar och 14 kor.

Formatsträng: "Totalt %d hästar och %d kor"

Omvandlingsspecifikationer: %d - d anger heltal

```
float vinkel 1.23456;
```

```
printf("%5.2f",vinkel);
```

f anger flyttal, decimaltal

5 är vidden på platsen för talet (om den räcker)

2 är antalet decimaler

Pröva exempel nummer 3, i boken sidan 99

scanf

```
scanf("%d %f", &antal, &vikt);
```

Denna låter oss mata in två värden i två olika variabler, ett heltal i variabeln antal och ett flyttal i variabeln vikt.

20 4.67 <enter> om man vill mata in 20 i antal och 4.67 i vikt.

&-tecknet – adressoperator, skickar adressen i minnet till variabeln istället för värdet på variabeln till funktionen scanf.

-Detta då vi vill fylla den med ett nytt värde snarare än att veta vad den har för värde.

Delbarhet

```
int i = 123;  
if (i%3==0)  
    printf("Talet är delbart med 3");  
else  
    printf(" Talet är ej delbart med 3 ");
```

Slumptal

*Till C standarden följer ett antal bibliotek med en massa användbara funktioner. I biblioteket **stdlib.h** hittar vi en pseudo-slumptalsgenerator:*

- ***srand(234);**
skickar "fröet" 234 till generatorn*
- ***rand();**
beräknar nästa slumptal i den serie som utgår från fröet
slumptalet är ett av heltalen mellan 0 och **RAND_MAX-1**
RAND_MAX kan skrivas ut och är typiskt 32767*

Slumptalsexempel

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void){
    srand(time(0)*time(0));
    int i;
    for(i=0;i<10;i++)
    {
        printf("%d\n", rand()%13+1);
    }
}
```

Är alla resultat
lika sannolika?

Exempelresultat:

2

4

12

1

9

13

8

5

9

1

math.h

I biblioteket **math.h** hittar vi flera användbara matematikfunktioner bla:

- Trigonometri: *cos, sin, tan, acos, asin, atan*
- *e* och naturliga logaritmen: *exp, log*
- Upphöjt i och roten ur: *pow, sqrt*

- $c = a^b$ alt1 : $c = \exp(b * \log(a));$ ty $a^b = e^{\ln a^b} = e^{b \ln a}$
 alt2: $c = \text{pow}(a,b);$