

KTH  
Matematik  
Per Enqvist

## Introduktion till Matlab

- trigonometri och grafritning

Programmet startas genom att skriva `matlab` i ett terminalfönster. Eventuellt kan det även startas genom något menysystem.

Ni kommer efter att programmet startat se en matlab prompt

```
>>
```

Här kan man nu skriva sina kommandon. Skriv t.ex.

```
>> cos(pi)
```

och tryck enter, så bör ni nu få svaret  $-1$ .

Som synes så antar funktionen `cos` att argumentet ges uttryckt i radianer, så har man en vinkel uttryckt i grader måste man alltså först omvandla denna till en vinkel uttryckt i radianer.

Om man vill bestämma cosinus för 90 grader får man alltså skriva:

```
>> cos(90/180*pi)
```

Om man ska använda en vinkel flera gånger kan det vara bra att införa en variabel.

```
>> v=90/180*pi
```

definierar en variabel `v`, och vi kan därefter lätt använda den, t.ex. för att definiera nya variabler

```
>> viktigt_uttryck = sin(v)+cos(v)/tan(2*v+1);
```

Det senaste kommandot avslutades med ett semikolon (;), dess funktion är att säga till matlab att inte skriva ut ett svar när man trycker på enter.

Ett viktigt användningsområde för Matlab är att rita funktionsgrafer, och andra bilder. Man kan öppna ett grafikfönster genom att skriva

```
>> figure
```

Bara för att åskådliggöra ritningsmöjligheterna kan vi skriva in följande kommandon

```
>> v=0:0.1:2*pi, plot(cos(v),sin(v)), axis square
```

varvid en enhetscirkel ritas upp. Det första kommandot skapar en vektor med vinklar från 0 till  $2\pi$ , och det andra ritar upp cosinus och sinus för dessa vinklar. För att få reda på vad det sista kommandot gjorde kan ni använda matlabs hjälpfunktion och skriva

```
>> help axis
```

varvid en förklarande text om kommandot visas.

Vi börjar nu om med en ny graf, skriv `clf` vid kommandoraden för att sudda den förra grafen.

Nu ska vi rita upp en fyrhörning, skriv in

```
>> x1=0; x2=2; x3=1.5; x4=0.5;  
>> y1=0; y2=0; y3=1; y4=1;  
>> patch([x1 x2 x3 x4 x1],[y1 y2 y3 y4 y1], 'r')
```

skriv `hold on` vid kommandoraden för att låsa fast grafen med fyrkanten. Vi kan nu rita upp en triangel i samma fönster:

```
>> x1=0; x2=2; x3=1;  
>> y1=0; y2=0; y3=1;  
>> patch([x1 x2 x3 x1],[y1 y2 y3 y1], 'y')
```

Nu känner ni till de flesta kommandon ni behöver, men för att slippa skriva om kommandon kan det vara bra att känna till hur man kan skriva en liten programfil. Skriv `edit` vid prompten så öppnas en fileditor. Skriv in några av kommandona vi använt tidigare där och spara filen som `test.m`. När ni nu skriver `test` vid prompten så utförs alla kommandon som står i er fil. Ni kan notera att om det står ett procenttecken, `%`, på en rad så utförs inget som står efter detta på raden. På hemsidan finns det en fil, `kub.m`, som ni kan ladda ner och prova om ni vill.

De som vill veta mera kan prova matlabs egna demon, dessa startas genom att skriva `demo matlab` eller `demo matlab graphics`.