

Mat-1.C Matemaattiset ohjelmistot

Maaliskuu 2012

HA

Document mode vs. worksheet mode.

Document: Uudempi,tavoitteena tyylik"as matemaattinen dokumentti/laskenta-arkki

Worksheet: Korostaa enemm"an laskentaa. Tyylik"as sek in matemaattisine kaava-edtointeineen.

Varsinainen **toiminta alkaa kohdasta 6**, lue kuitenkin ainakin 1 läpi huolella.

Maple 15

Oppilaat saavat asentaa omalle koneelleen, lisenssi voimassa vuoden loppuun.

Pikainen johdanto

Pieni alkukokeilu: Käytetään hiiren oikeaa, "context sensitive", jolloin ei tarvitse tietää komentojen nimiä.

> $(x + y)^5$ # Hiirell" a siniseen, oikealla "context sensitive" -> expand

$$(x + y)^5 \quad (2.1)$$

> expand((2.1))

$$x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + y^5 \quad (2.2)$$

> factor((2.2))

$$(x + y)^5 \quad (2.3)$$

> diff((2.3), x)

$$5(x + y)^4 \quad (2.4)$$

> eval((2.4), [x = 1, y = 2])

$$405 \quad (2.5)$$

Yhtälön ratkaiseminen:

> $a \cdot x^2 + b \cdot x + c = 0;$

$$ax^2 + bx + c = 0 \quad (2.6)$$

> solve({ (2.6) }, [x])

$$\left[\left[x = \frac{1}{2} \frac{-b + \sqrt{b^2 - 4ac}}{a} \right], \left[x = -\frac{1}{2} \frac{b + \sqrt{b^2 - 4ac}}{a} \right] \right] \quad (2.7)$$

Visualisointia

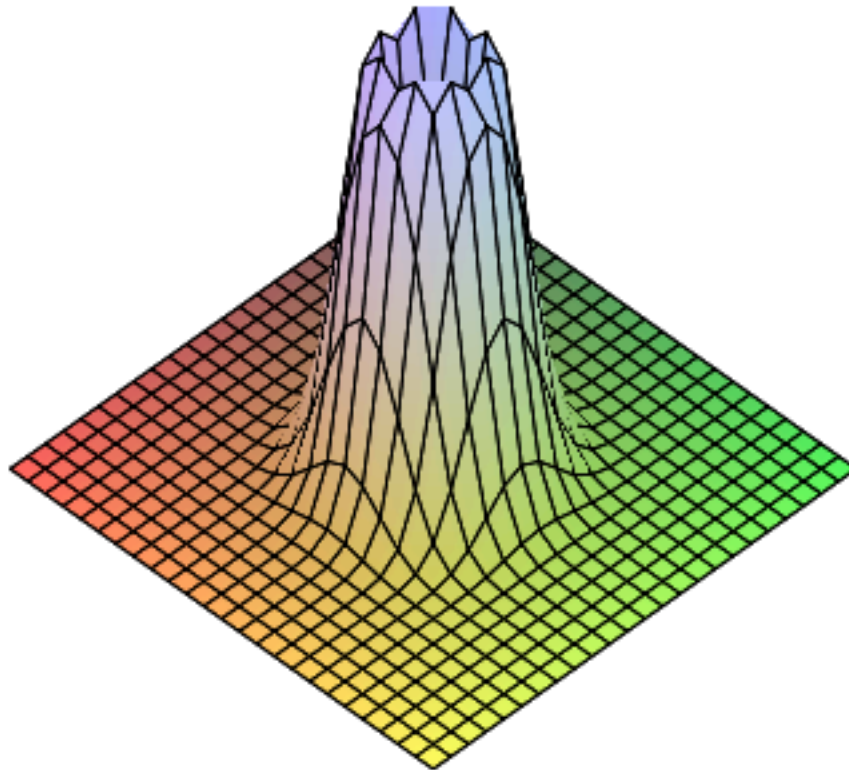
> $(x^2 + y^2) \cdot \exp(-x^2 - y^2)$

Klikkaa taas sinist" a kaavaa. Py"orittele kuvaa, valitse esim. Color > z-> color HUE

$$(x^2 + y^2) e^{-x^2 - y^2}$$

(2.8)

```
> smartplot3d[x, y]( (2.8) )
```



1. Työskentelyohjeita, työarkin käsittelyn perusasiat

Muista tallettaa aika ajoin (CTR-S) tai File-valikko: save

[Mac-työskentelyn yleisohje: CTR -> CMD]

Talletus on joka tapauksessa syytä tehdä aina ennen jotain potentiaalisesti isoa laskentaa.

Tämä työarkki sisältää ohjeita ja ohjeisiin liittyviä tehtäviä, joita voi samantien ryhtyä kokeilemaan. Voit tehdä muistiinpanoja työarkille ja tallettaa sen itsellesi.

Suositus: Selaa työarkkia jonkin matkaa, totuttele heti helpin käyttöön.

Harjoitustehtävädokumentti:

Kannattaa aloittaa INSERT-valikon SECTION-valinnalla ja napsauttaa muutama sektio heti kättelyssä arkille.

Huomaa, että työarkin voi julkaista (export) my"os pdf tai HTML-muodossa. Edellinen siistimpi, jälkimmäisessä toimii my"os animaatiot.

Elämänohje:

Maplen filosofiaa kannattaa opetella sen verran, että työskentely käy nautittavaksi. (Sama pätee Matlabiin (ja Mathematicaan).)

Tavoitteita:

- Työarkin käsittely
- Peruslaskutoimitukset ja sievennykset
- Grafiikka
- Vapaat ja sidotut muuttujat
- Lauseke vs. funktio
- Matriisit
- Perustietorakenteet, erit. **jonot, joukot, listat.**
- **Vertailu Matlab-työskentelyyn**

Katsotaan ensin työarkkiin (worksheet) liittyviä asioita.

- Uuden (laskenta)kehotteen saa hiirellä työkalunauhasta [>] tai CTR-J (jälkeen) tai CTR-K (ennen)
- Tekstitila: **T** (edellisen vasemmalla puolella) tai CTR-t (Mac: CMD-t)
- **F5 Math/Input ja text/math**
- Math-tilassa matemaattista kaavaa voi editoida k"atev"asti. Nuoli oikealla -> vie takaisin perustasolle.

T"ass"a F5 ja sitten vain editoidaan kaava, kokeile!

$$\frac{x^2 + y^2}{x^y}$$

Osoittajaa ei tarvitse suluttaa, maalataan ja sitten jakomerkki /

- Uusi luku: INSERT-valikko, ->SECTION
- Ei tarvitse menn"a rivin loppuun ennen ENTER:ia.
- CTR-l (label) liitt"a"a viittauksen aiemman tuloksen kaavanumeroon.

Yleisiä ohjeita:

Leikkaa/liimaa: UNIX/X:ssä oikein kätevä, eli kuten aina X:ssä

- Maalataan hiiren vasemmalla
- viedään kursori haluttuun kohtaan

- liimataan keskimmaisella

(Windows:ssa maalaus, CTR-C, vienti, CTR-V, Mac: CTR <-> CMD)

Ennen "lopullista" tallettamista kannattaa valita EDIT-valikosta **remove output** => tiedosto pieneksi.

Käänteinen toimenpide on EDIT-valikon edellinen valinta: **execute worksheet**.

Matemaattista tekstinkäsittelyä Maplella

Versiossa 15 varsin kehittynyt. Ideaali: Matemaattisin kaavoin, kuvin ym. varustettu dokumentti, joka suorittaa itsensä.

Toki LaTeX on matemaattisen tekstinkäsittelyn ykkönen. Maplella k"atevaa generoida LaTeX-kaavoja. komennolla **latex**.

Omia kokeiluja ... tekstia, matem. kaavoja...

Seuraavissa voit kokeilla laskentaa. Tee lis"a" a kehoitteita CTR-J.

2. Viitteit"a, ty"oarkkeja, koodeja

Samat linkit ovat luentosivulla.

[HAM] Apiola: Symb ja num. lask Maple-ohjelmalla, Otatieto 588

Tämä teos ilmeisine puutteineenkin lienee hyödyllinen. Perustuu versioon 5.

Täydennystä vaatii erityisesti matriisilaskentaosuus, sillä uusi **LinearAlgebra**-tyyli tuli vasta versioon 6.

Kirjastossa useita, jos haluat omaksi, saat minulta alennushintaan (10 e), vain muutama j"aljella.

Kurssin Maple-hakemisto: -- MUUTA-- www.math.hut.fi/teaching/numsym/05/maple/

[PIKA] Apiola-Peltola: Maple-pikaopas <http://www.math.hut.fi/teaching/k3/maple-pikaopas.html>

[SOL] Solmun Maple-kirjoitus <http://www.math.helsinki.fi/Solmu/solmu12/apiola>

[KOF] M. Kofler: Maple, an introduction and reference, Addison Wesley 1997 www.awl-he.com, erittäin hyvä Maple-yleiskirja.

[LYN] Lynch: Dynamical systems with Applications using Maple, Birkhäuser (Mat-kirjastossa) (Maple-tutoriaalit ja paljon diffyhtälökoodeja) www.maplesoft.com/apps/

[HECK] Heck: An Introduction to Maple (Springer), täydellisin ja kattavin Maple-kirja

[ISR] Israel: Calculus with Maple (pari ylim. kappaletta kirjastossa)

Israel: Maple advisor database (kts. linkki luentosivulla)

[Coomb] Coombes-Hunt-Lipsman-Osborn-Stuck: Diff. Equations with Maple (ainakin 1 kirjastossa)

Kurssia varten kirjoitettuja/tettavia Maple-työarkkeja, kuten LA, ominaisarvot, lindys, fourier, numint, ...[maple/ns05.mpl](#)

3. Opiskelusuositus

[Käy samalla läpi [HAM]-kirjan ss. 13 - 51 (Alkutoimenpiteet)

4. Työarkin (worksheet) lataaminen

Kun lataat työarkin FILE-valikon OPEN-valinnalla, saat käyttöösi visuaalisen esityksen Maple-työstä.

Työarkilla olevat **komennot suorittuvat vasta**, kun siirryt punaiseen INPUT-soluun ja **painat ENTER:iä**.

Koko työarkin kaikki komennot saat suoritetuksi **Edit valikon** valinnalla "**execute worksheet**", kuten edellä todettiin.

Jos haluat jatkaa kokeiluja tällä kohdalla, tee lisää kehoitteita joko klikkaamalla kohtaa > tai CTR-J (jos jälkeen) (CTR-K (ennen))

```
> 247*3756;
                                     927732
(6.1)
```

```
> Pi; evalf(Pi, 50);
                                     π
3.1415926535897932384626433832795028841971693993751
(6.2)
```

```
> pi; evalf(pi);
# Huomaa: Pi on muuttuja, jolla on arvo 3.14..., pi on vain tekstink"asittelyyn (Maplesoftin
kiusa?)
                                     π
                                     π
(6.3)
```

5. Laskentoa numeroilla

```
> restart: # Työn alkuun kannattaa yleensä laittaa.
```

Laskuoperaatiot ovat "normaalit" $+ - * / ^$, **kertomerkkiä ei saa jättää** pois (kuten Mathematicassa)

Kokeile jotain, voit myös sijoittaa muuttujiin tyyliin:

```
> oso:=123; nimi:=-456; luku:=oso/nimi;
      oso := 123
      nimi := -456
      luku := -  $\frac{41}{152}$ 
(7.1)
```

Maple vaatii loppumerkin, joko ; tai : Kokeile, jos et jo tiedä.

Lue lisää:

[PIKA] s 1.3 s. 6

[SOL] s. 8

[HAM] s. 21

6. Laskentoa symboleilla ja numeroilla

Vapaat ja sidotut muuttujat

Lue

[SOL] s 15-16 Symbolien hallintaa ja periaatteita

[HAM] s. 70 - 71 (korjaus: <http://www.math.hut.fi/~apiola/maple/opus/eval.html>)

(see Heck Ch 3 Variables and names p. 65 -..

The secret behind success of CA : **free (unbound) variables** . .)

On hyvä totutella laittamaan **restart** työarkin alkuun (ja muihinkin hengähdyspaikkoihin)
Painele nyt ENTER:iä ja palaa ohjeen mukaan takaisin.

```
> restart;  
> lauseke:=(a+b)^2;          # Muuttujat a ja b ovat "vapaat"  
                             lauseke := (a + b)2                                (8.1)
```

```
> lauseke:=expand(%);  
                             lauseke := a2 + 2 a b + b2                                (8.2)
```

```
> a:=x^2 :b:=exp(c*z):  
> lauseke;  
                             x4 + 2 x2 ecz + (ecz)2                                (8.3)
```

```
> x:=1:  
> lauseke;  
                             1 + 2 ecz + (ecz)2                                (8.4)
```

```
> s := b;  
                             s := ecz                                            (8.5)
```

Tämä on ensimmäinen merkittävä havainto: Kun x muuttuu, niin **lauseke** muuttuu, koska yllä olevassa sijoituslauseessa muuttujat a ja b olivat vapaat.

Täyseyvaluatio

Warning, inserted missing semicolon at end of statement

```
                             1 + 2 ecz + (ecz)2                                (8.6)
```

Jos haluat vapauttaa vain joitakin valittuja muuttujia, niin:

Muuttujien vapauttaminen arvostaan:

Joko **a:='a'**; tai **unassign('a')**:

```
> a:='a': # Vapautetaan a  
> lauseke;  
                             a2 + 2 a ecz + (ecz)2                                (8.7)
```

```
> b:='b': lauseke;  
                             a2 + 2 a b + b2                                (8.8)
```

Filosofiesimerkki täyseyvaluatation havainnollistukseen:

```
> Sokrates := ihminen;      # ihminen on vapaa, kun Sokrateelle annetaan arvo.  
   ihminen := kuolevainen;  
                             Sokrates := ihminen  
                             ihminen := kuolevainen                                (8.9)
```

```
> Sokrates;  
                             kuolevainen                                (8.10)
```

```
> ihminen := kuolematon; # Muutetaan ihmisen arvoa.
                               ihminen := kuolematon (8.11)
```

```
> Sokrates;
                               kuolematon (8.12)
```

Jos ihminen on sidottu, kun Sokrateelle annetaan arvo, niin Sokrateen arvoksi tulee ihmisen arvo (ei ihminen).

```
> unassign('Sokrates'); unassign('ihminen');
> Sokrates; ihminen;
                               Sokrates
                               ihminen (8.13)
```

```
> ihminen := kuolevainen;
                               ihminen := kuolevainen (8.14)
```

```
> Sokrates := ihminen;
                               Sokrates := kuolevainen (8.15)
```

```
> Sokrates; # Niinpä.
                               kuolevainen (8.16)
```

```
> ihminen := kuolematon;
                               ihminen := kuolematon (8.17)
```

```
> Sokrates; # Arvo ei muutu, sillä Sokrateella ei ole mitään jälkeä 'ihmisestä'.
                               kuolevainen (8.18)
```

Tietysti Sokrates muuttuu, jos 'kuolevainen' muuttuu.

```
> kuolevainen := Pi;
                               kuolevainen :=  $\pi$  (8.19)
```

```
> Sokrates;
                                $\pi$  (8.20)
```

Täysevaluaatio on yksi tärkeä ominaisuus symbolilaskennan polulla.

Vakiot Pi ja I

```
> Pi, I;
                                $\pi, I$  (8.21)
```

```
> evalf(Pi);
                               3.141592654 (8.22)
```

```
> evalf(Pi, 30);
                               3.14159265358979323846264338328 (8.23)
```

```
> 2/3+4/5; evalf(%);
                                $\frac{22}{15}$ 
                               1.466666667 (8.24)
```

```
> evalf(pi);
                               (8.25)
```

π

(8.25)

Huomaa tämä: Maple osaa vain "tekstinkäsittelä" π iä, laskentaan tarvitaan Pi
Tässä tulee usein virheitä alkavalle Maple-urheilijalle. (Voidaan kysyä, miksi Maplesoft haluaa tällain
kohden kiusata käyttäjää.)

Virhetilanteista yleensä:

[HAM] Liite A s. 191 -198, π -problematiikka s. 192

[PIKA] Lopussa lyhyt lista (jossa myös samainen π)

Kompleksilukuja, imag. yksikkö i

Kompleksisievennys, erityisesti *evalc*

evalc

(8.26)

> $(1+2i)/(2-3i)$; # Numeerinen kompleksiluku muuntuu
automaattisesti muotoon $x+iy$.

$$-\frac{4}{13} + \frac{7}{13}i$$

(8.27)

Jos kompleksiluku annetaan symbolisessa muodossa, Maple olettaa a :n ja b :n reaaliksi esityksessä
 $a+ib$.

Warning, inserted missing semicolon at end of statement

 $aba + ib$

(8.28)

> $z1:=a1+i*a2; z2:=a2+i*b2;$

$$z1 := a1 + ia2$$

$$z2 := a2 + ib2$$

(8.29)

> $z1/z2;$

$$\frac{a1 + ia2}{a2 + ib2}$$

(8.30)

Komento *evalc* on varsin tehokas, kannattaa muistaa kompleksisievennyksissä.

Warning, inserted missing semicolon at end of statement

evalc

(8.31)

> *evalc*(%);

evalc

(8.32)

> **abs** ja **argument**:

> $z := ((1+2i)/(3-4i));$

$$z := -\frac{1}{5} + \frac{2}{5}i$$

(8.33)

> $r := \text{abs}(z);$

$$r := \frac{1}{5}\sqrt{5}$$

(8.34)

> $\phi := \text{argument}(z);$

$$\phi := -\arctan(2) + \pi$$

(8.35)

> $r \cdot \exp(i \cdot \phi);$

$$\frac{1}{5}\sqrt{5} e^{i(-\arctan(2) + \pi)}$$

(8.36)


```
> evalc(%);
```

$$-\frac{1}{5} + \frac{2}{5}i \quad (8.37)$$

Esim: 1:n juuret

```
> restart;
```

```
> w:=exp(I*2*k*Pi/n);
```

$$w := e^{\frac{2Ik\pi}{n}} \quad (8.38)$$

```
> with(plots): # Ladataan plots-kirjastopakkaus.
```

```
> setoptions(scaling=constrained,axes=framed): # Sama skaala  
akseleilla
```

```
> n:=5: ykkosen5juuret:=seq(w,k=0..n-1);
```

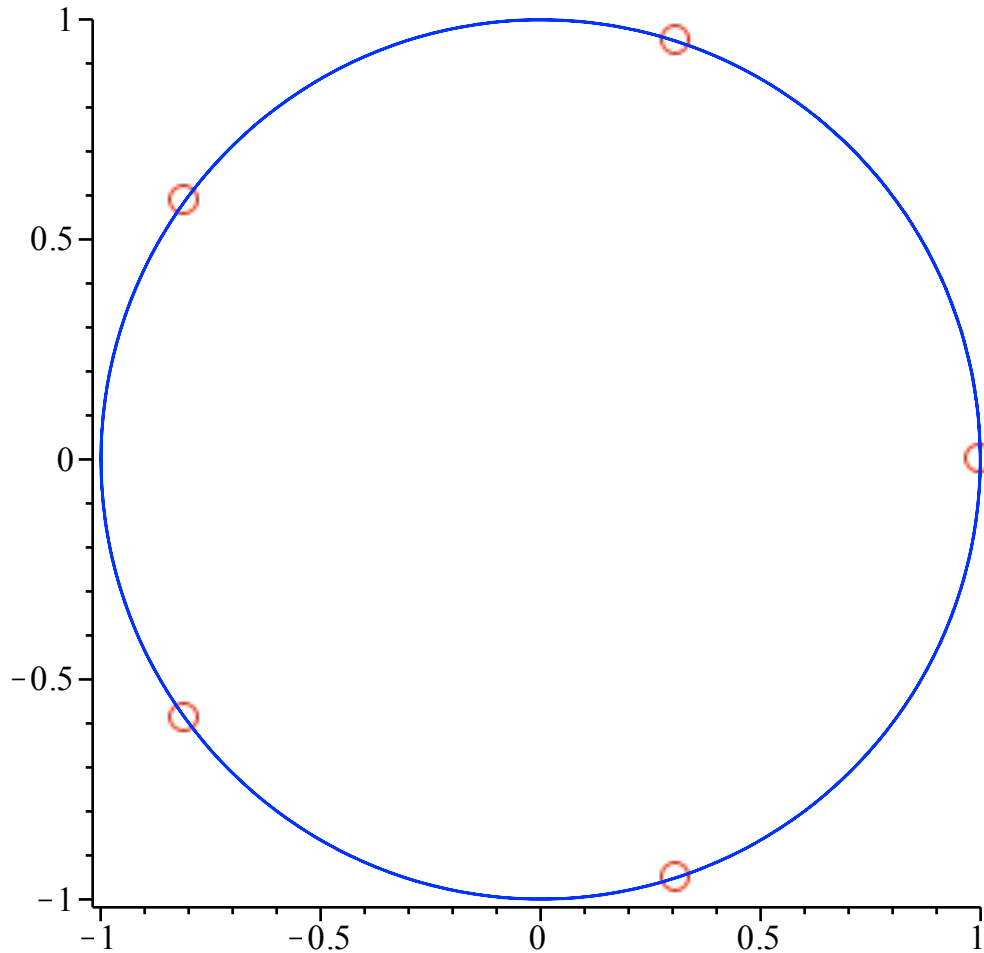
$$ykkosen5juuret := 1, e^{\frac{2}{5}i\pi}, e^{\frac{4}{5}i\pi}, e^{-\frac{4}{5}i\pi}, e^{-\frac{2}{5}i\pi} \quad (8.39)$$

```
> rinkulat:=complexplot([ykkosen5juuret],style=point,symbol=circle,  
symbolsize=20):
```

```
> yksikkoymp:=complexplot(exp(I*Pi*t),t=0..2*Pi,color=blue):
```

Grafiikat saadaan samaan kuvaan funktiolla **display** (joka asustaa **plots**-pakkauksessa), kts
alempana.

```
> display(rinkulat,yksikkoymp);
```



Huom: **complexplot** tuli vasta versiossa 7, siksi [HAM]:ssa esitettävät pikku konversiotempot eivät ole enää tarpeen kompleksipiirron yhteydessä.

[Warning, inserted missing semicolon at end of statement](#)
plots:-complexplot

(8.40)

Vrt: Matlab:

```
>> n=5;
>> K=0:n-1;
>> w=exp(2*K*i*pi/n);
>> plot(w,'or')
>> hold on
>> t=linspace(0,2*pi); plot(exp(i*t),'b')
>> axis equal
```

▼ 7. Jono, joukko, lista, taulu(kko),vektori

Lue: [HAM] s. 30 ja ss. 76 - 84

▼ **Joukot { }, jonot, listat []**

$$f(a, b, c) \quad (9.1.9)$$

Huomaa, että jono ei ole samanlainen "paketti" kuin lista tai joukko. map-esimerkki osoittaa, että voidaksemme soveltaa tällaista operaatiota jonoon, se on ensin ympäröitävä listasuululla `[]` ja lopuksi riisuttava sulut pois tuloksesta `op`-komennolla:

`Error, missing operator or `;``

```
> op(map(f, [jono]));
```

$$f(a), f(b), f(c) \quad (9.1.10)$$

Funktion soveltaminen listan tms. kaikkiin alkioihin, map

Matlabissa kaikki ns. "skaalaarifunktiot", kuten kaikki matemaattiset 1:n muuttujan funktiot toimivat automaattisesti alkioittain

vektoreihin ja matriiseihin. Siten `sin([1,2,3])` antaa `[sin(1),sin(2),sin(3)]`.

Maplessa näin ei ole, mutta se saadaan aikaan "map"-lla:

```
> x:=[a,b,c]; fx:=map(f,x);
```

$$\begin{aligned} x &:= [a, b, c] \\ fx &:= [f(a), f(b), f(c)] \end{aligned} \quad (9.2.1)$$

```
> restart;
```

```
> M:=Matrix([seq([seq((x+y)^(i+j),j=1..3)],i=1..2)]);
```

$$M := \begin{bmatrix} (x+y)^2 & (x+y)^3 & (x+y)^4 \\ (x+y)^3 & (x+y)^4 & (x+y)^5 \end{bmatrix} \quad (9.2.2)$$

```
> map(expand,M);
```

$$\begin{aligned} & [[x^2 + 2xy + y^2, x^3 + 3x^2y + 3xy^2 + y^3, x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4], \\ & [x^3 + 3x^2y + 3xy^2 + y^3, x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4, x^5 + 5x^4y \\ & + 10x^3y^2 + 10x^2y^3 + 5xy^4 + y^5]] \end{aligned} \quad (9.2.3)$$

```
> expand~(M);
```

$$\begin{aligned} & [[x^2 + 2xy + y^2, x^3 + 3x^2y + 3xy^2 + y^3, x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4], \\ & [x^3 + 3x^2y + 3xy^2 + y^3, x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4, x^5 + 5x^4y \\ & + 10x^3y^2 + 10x^2y^3 + 5xy^4 + y^5]] \end{aligned} \quad (9.2.4)$$

Hiukan perusteellisemmin (osittain toistaen)

[Lue: [HAM] s. 30 ja ss. 76 - 84

```
> restart: with(LinearAlgebra):
```

```
> jono1:=a,b,c; jono2:=seq(x^i,i=-3..3);
```

$$\begin{aligned} jono1 &:= a, b, c \\ jono2 &:= \frac{1}{x^3}, \frac{1}{x^2}, \frac{1}{x}, 1, x, x^2, x^3 \end{aligned} \quad (9.3.1)$$

```
> lista1:=[jono1]; lista2:=[jono2];
```

$$\begin{aligned} lista1 &:= [a, b, c] \\ lista2 &:= \left[\frac{1}{x^3}, \frac{1}{x^2}, \frac{1}{x}, 1, x, x^2, x^3 \right] \end{aligned} \quad (9.3.2)$$

```
> joukko1:={jono1}; joukko2:={jono2}; # Alkioiden järjestys on "satunnainen"
```

$$\begin{aligned} \text{joukko1} &:= \{a, b, c\} \\ \text{joukko2} &:= \left\{1, x, \frac{1}{x^3}, \frac{1}{x^2}, \frac{1}{x}, x^2, x^3\right\} \end{aligned} \quad (9.3.3)$$

```
> joukko1 union {a,b,-1,1};
      {-1, 1, a, b, c} (9.3.4)
```

```
> [op(lista1),op(lista2)]; # Listat liitetään liittämällä
operandijonot ja ympäröimällä []:lla
      [a, b, c, 1/x^3, 1/x^2, 1/x, 1, x, x^2, x^3] (9.3.5)
```

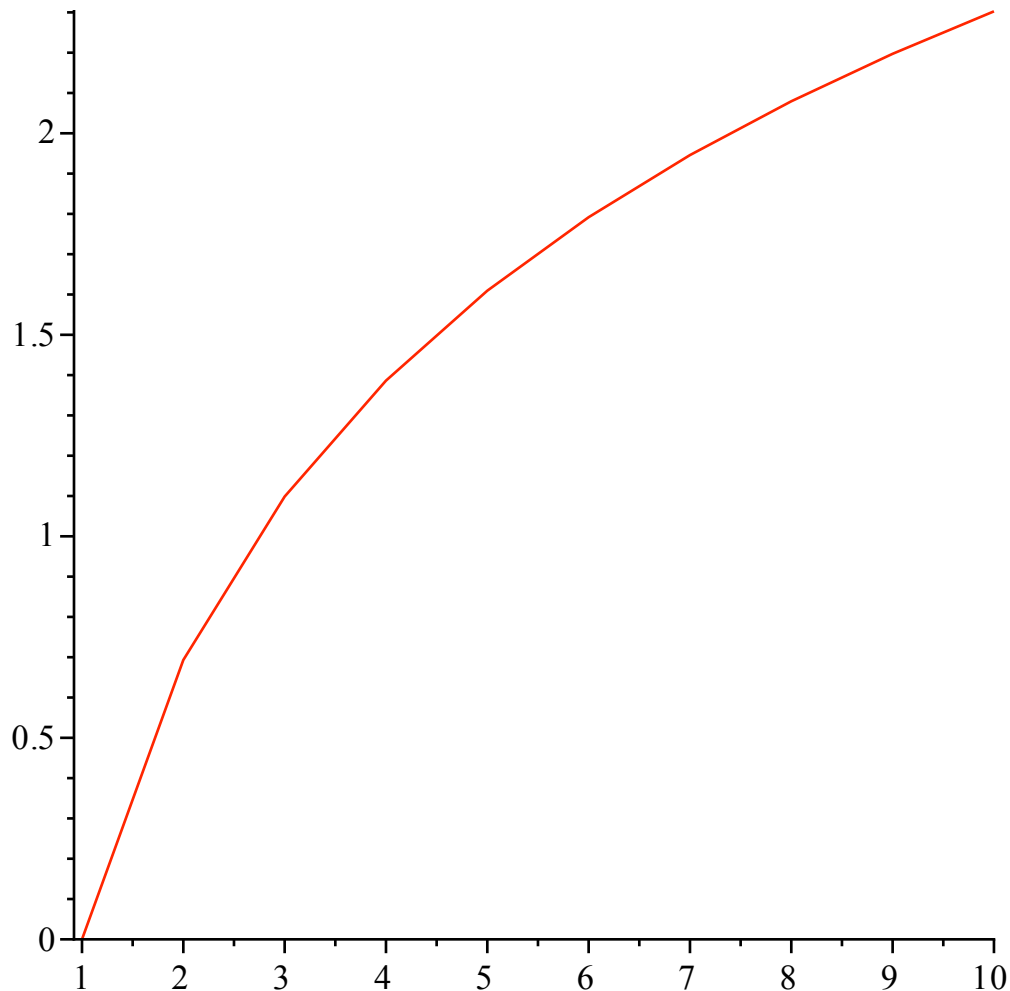
```
> logaritmit:=seq([i,log(i)],i=1..10);
logaritmit := [1, 0], [2, ln(2)], [3, ln(3)], [4, 2 ln(2)], [5, ln(5)], [6, ln(6)], [7,
ln(7)], [8, 3 ln(2)], [9, 2 ln(3)], [10, ln(10)] (9.3.6)
```

Maplessa vaakavektoreiden lista ei ole sama kuin matriisi (kuten Mathematicassa). Se voidaan muuntaa **Matrix:lla**, takaisin saadaan **convert:lla**.

Esitys listojen listana on hyödyllinen esimerkiksi piirrettäessä, se muoto kelpaa suoraan **plot:lle**.

```
> logtaulu:=Transpose(Matrix([logaritmit]));
logtaulu := [ 1  2  3  4  5  6  7  8  9  10
              0 ln(2) ln(3) 2 ln(2) ln(5) ln(6) ln(7) 3 ln(2) 2 ln(3) ln(10) ] (9.3.7)
```

```
> plot([logaritmit]);
```



```
> convert(logtaulu,listlist);
[[1, 2, 3, 4, 5, 6, 7, 8, 9, 10], [0, ln(2), ln(3), 2 ln(2), ln(5), ln(6), ln(7), 3 ln(2),
  2 ln(3), ln(10)]] (9.3.8)
```

```
> convert(Transpose(logtaulu),listlist);
[[1, 0], [2, ln(2)], [3, ln(3)], [4, 2 ln(2)], [5, ln(5)], [6, ln(6)], [7, ln(7)], [8,
  3 ln(2)], [9, 2 ln(3)], [10, ln(10)]] (9.3.9)
```

Vektorit ja matriisit:

```
> restart;
> pysty:=<a,b,c>; vaaka:=<1 | 2 | 3>;
      pysty :=  $\begin{bmatrix} a \\ b \\ c \end{bmatrix}$ 
      vaaka :=  $\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$  (9.3.10)
```

```
> M:=<<1,2,3>|<b,e,x>|<u,v,w>>; # Sarakkeittain.
      M :=  $\begin{bmatrix} 1 & b & u \\ 2 & e & v \\ 3 & x & w \end{bmatrix}$  (9.3.11)
```

```
> Mt:=<<1|2|3>,<b|e|x>,<u|v|w>>; # Riveittäin
      Mt :=  $\begin{bmatrix} 1 & 2 & 3 \\ b & e & x \\ u & v & w \end{bmatrix}$  (9.3.12)
```

```
> M.Mt; # Matriisikertolasku
       $\begin{bmatrix} 1 + b^2 + u^2 & 2 + b e + u v & 3 + b x + u w \\ 2 + b e + u v & 4 + e^2 + v^2 & 6 + e x + v w \\ 3 + b x + u w & 6 + e x + v w & 9 + x^2 + w^2 \end{bmatrix}$  (9.3.13)
```

```
> convert(vaaka,list);
      [1, 2, 3] (9.3.14)
```

```
> convert(pysty,list);
```

```
> convert(M.Mt,listlist);
      [a, b, c]
[[1 + b^2 + u^2, 2 + b e + u v, 3 + b x + u w], [2 + b e + u v, 4 + e^2 + v^2, 6 + e x
  + v w], [3 + b x + u w, 6 + e x + v w, 9 + x^2 + w^2]] (9.3.15)
```

Tehtäviä

Miten poistaisit listasta saman alkion toistot?

Ratk.

Otetaan esimerkki:

```
> restart;
> L:=sort([seq(seq(i,i=1..k),k=1..9)]);
L := [1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 5,
  5, 5, 5, 5, 6, 6, 6, 6, 7, 7, 7, 7, 8, 8, 9] (9.4.1.1)
> J:={op(L)}; # Muutetaan lista joukoksi => toistot
```

```

poistuvat!
J := {1, 2, 3, 4, 5, 6, 7, 8, 9} (9.4.1.2)
> Lriisuttu := [op(J)]; # Takaisin listaksi
Lriisuttu := [1, 2, 3, 4, 5, 6, 7, 8, 9] (9.4.1.3)
> sort(Lriisuttu); # Joukoksi muuntaminen saattaa muuttaa
alkioiden järjestyksen.
[1, 2, 3, 4, 5, 6, 7, 8, 9] (9.4.1.4)
Tietysti voidaan ottaa alkioita, joilla ei ole järjestystä, silloin sort ei tee mitään.
> L := [seq(seq(x[i], i=1..k), k=1..6)];
L := [x1, x1, x2, x1, x2, x3, x1, x2, x3, x4, x1, x2, x3, x4, x5, x1, x2, x3, x4, x5, x6] (9.4.1.5)
> {op(L)};
{x1, x2, x3, x4, x5, x6} (9.4.1.6)
> [op(%)];
[x1, x2, x3, x4, x5, x6] (9.4.1.7)

```

8. Symbolinen lauseke vs. funktio

Tämä on tärkeä asia ymmärtää kunnolla.

[SOLmu] s. 17 Ohjelmointi

[HAM] 2.4 Matemaattiset funktiot s. 60 - 63

Israel: Advisor database ..maple/advisor/fundef.mws

Pohdiskelua

Matemaattisen funktion käsittely **Maple-lausekkeena**

```

> f := x^2;
f := x^2 (10.1.1)

```

Määrittelemme muuttujan f, jota käytämme matemaattisen funktion tavoin. Maplen kannalta **f on muuttuja**, jonka arvoksi olemme sijoittaneet lausekkeen x^2

Jos haluamme laskea lausekkeen arvon eri x:n arvoilla, joudumme käyttämään subs-komentoa.

```

> subs(x=5, f);
25 (10.1.2)

```

```

> eval(f, x=5); # Näinkin voidaan, tätä tapaa käytetty gsg:ssä.
25 (10.1.3)

```

Jos haluat seurata lauseke/funktio-juonta, hyppää tässä kohdassa seq-komentojen yli.

```

> seq(subs(x=k, f), k=0..10);
0, 1, 4, 9, 16, 25, 36, 49, 64, 81, 100 (10.1.4)

```

```

> a:=-4: b:=4: N:=10: h:=(b-a)/N: arvojono:=seq(subs(x=a+k*h,
f), k=0..N);
arvojono := 16, 256/25, 144/25, 64/25, 16/25, 0, 16/25, 64/25, 144/25, 256/25, 16 (10.1.5)

```

```

> evalf(arvojono);
16., 10.24000000, 5.760000000, 2.560000000, 0.6400000000, 0., 0.6400000000,
2.560000000, 5.760000000, 10.24000000, 16. (10.1.6)

```

VAROITUS: Kun matemaattista funktiota käsitellään lausekkeena, on oltava johdonmukainen.

Siitä ei saa välillä käyttää merkintää $f(x)$, tietenkään!

Kokeile vaikka:

```
> f(x); f(5);
```

$$\begin{aligned} &x(x)^2 \\ &x(5)^2 \end{aligned}$$

(10.1.7)

Matemaattisen funktion käsittely **Maple-funktiona**:

```
> f := x -> x^2;
```

$$f := x \rightarrow x^2$$

(10.1.8)

```
> f(x), f(y), f(5);
```

$$x^2, y^2, 25$$

(10.1.9)

Tyytykäämme ensialkuun tähän. Lue kuitenkin jossain vaiheessa loppukin.

unapply on toinen funktion määrittelytapa.

```
> g := x^3;
```

$$g := x^3$$

(10.1.10)

```
> g(5); # Tämä on tuhoon tuomittu.
```

$$x(5)^3$$

(10.1.11)

Huomaamme nyt, että olis ollut mukavampi käsitellä funktiona. No sehän käy:

```
> g := unapply(g, x);
```

$$g := x \rightarrow x^3$$

(10.1.12)

```
> g(5);
```

$$125$$

(10.1.13)

What is the difference **unapply** vs **->** ?

Nuolimäärittely evaluoi (vasta) suoritusaikana.

unapply evaluoi (jo) määrittelyaikana.

```
> a:=x^2: f:=x->a;g:=unapply(a,x);
```

$$f := x \rightarrow a$$

$$g := x \rightarrow x^2$$

(10.1.14)

```
> f(3);
```

$$x^2$$

(10.1.15)

```
> g(3);
```

$$9$$

(10.1.16)

```
> a:=Pi;
```

$$a := \pi$$

(10.1.17)

```
> f(3);
```

$$\pi$$

(10.1.18)

```
> g(3);
```

$$9$$

(10.1.19)

Tässä tilanteessa siis nuolimäärittely **->**

tomii "omituisesti", kun taas **#unapply** toimii "odotetusti".

Siniset määrittelyrivit yllä kertovat kyllä asian hyvin, niitä kannattaa seuraila.

Jatketaan tästä (harj0):

Varoitus: **Hengenvaarallista: Älä koskaan määrittele funktiota tyyliin $F(x):=$ lauseke;** Tällöin tulee määrittelyksi funktion F arvo yhdessä pisteessä x , muilla symboleilla funktio f on määrittelemätön.

Poikkeus: Funktion määrittelyä täydentävien **poikkeusarvojen** määrittelyssä tämä on hyödyllistä, (esimerkki otetaan kohta.)

Palataan nyt vielä tuohon hengenvaaraan:

```
> restart;
```

```
> F(x):=x^3; # kokeillaan kiellettyä leikkiä.
```

$$F(x) := x^3 \quad (10.1.20)$$

Tuo on syntaksiltaan oikein, mutta hyvin hämäävää.

Mitä on $F(x)$

```
> F(x);
```

$$x^3 \quad (10.1.21)$$

Näyttää, kuin toimisi oikein. Vaan eipas toimi muilla kuin x :llä.

```
> F(x), F(3), F(a), F(sin(z));
```

$$x^3, F(3), F(a), F(\sin(z)) \quad (10.1.22)$$

Toisin sanoen funktio on määritelty vain, kun argumenttina on symboli x

Esimerkki: **Funktion määrittelyn täydentäminen poikkeusarvoilla.**

Haluamme määrittellä funktion, jota joskus kutsutaan nimellä **sinc**, olkoon tässä vain F

```
> F:=x->sin(x)/x;
```

$$F := x \rightarrow \frac{\sin(x)}{x} \quad (10.1.23)$$

```
> F(0);
```

Error, (in F) numeric exception: division by zero

Yleinen sääntö ei määrittele F -funktioita 0:ssa, koska siinä tulee 0:lla jako.

Annamme siksi funktiolle 0:ssa sopivan arvon suoralla asetuksella:

```
> F(0):=1;
```

$$F(0) := 1 \quad (10.1.24)$$

```
> F(a), F(1/2), F(0), F(sin(x));
```

$$\frac{\sin(a)}{a}, 2 \sin\left(\frac{1}{2}\right), 1, \frac{\sin(\sin(x))}{\sin(x)} \quad (10.1.25)$$

```
> seq(F(x), x=-2..2); evalf(%);
```

$$\frac{1}{2} \sin(2), \sin(1), 1, \sin(1), \frac{1}{2} \sin(2) \\ 0.4546487134, 0.8414709848, 1., 0.8414709848, 0.4546487134 \quad (10.1.26)$$

Remember: # **evalf** means: **evaluate floating point.**

Lue lisää:

[HAM] ss. 62-63

Harjoitustehtävä

Määrittele polynomifunktio $p(x) = x^3 - 4x^2 + 4x - 1$.

Määritä nollakohdat ja paikalliset minimi- ja maksimit. Piirrä funktio ja sen derivaatta.

Tarkista laskemalla funktion arvot, että nollakohdat ovat nollakohtia.

Käsittele polynomia ensin lausekkeena ja sitten funktiona. Mitkä ovat kunkin tavan hyvät/huonot

puolet.

Derivaattalauseke -- diff

Derivaatafunkti -- D (Käyttele helppiä)

?plot, ?solve, ?fsolve ?diff, ?D

Tämäntyyppisten tehtävien, niin yksinkertaisia matemaattisesti kuin ovatkin, sujuva hallinta tekee

Maple-työskentelystä nautittavaa puuhaa ja auttaa pitkälle.

10. LinearAlgebra, linalg, matriisit ja vektorit

Lue: www.math.hut.fi/~apiola/maple/opas/LA.pdf

[HAM]-kirjassa (kuten monissa muissakin) esitellään **lianlg**-tyyliä, joka on uuteen verrattuna luotaantyöntävä.

```
> with(LinearAlgebra):      # Tämä olkoon standardilatauskomentomme.
> #?LinearAlgebra          # Poista kommentti edestä, niin saat
    funktioluettelon ja selostusta.
```

11. Sievennystä ja yhtälön ratkaisua

Pohdiskelua

simplify - yleissieventäjä, ensimmäiseksi tarjoiltava.

expand - kertoo auki

collect - kokoaa

? collect ja kokeile joitakin esimerkkejä. Varsin hyvä komento monessa paikassa.

factor - Jakaa tekijöihin.

convert - ... Monenmoisiin konversioihin sievennyksessä esim **convert(lauseke, parfrac, muuttuja);**

Huomaa, että lauseke ei muutu, uusi tulos palautetaan. Jos halutaan päivittää lauseke, on komennettava

```
lauseke:=convert(lauseke, parfrac, muuttuja);
```

solve - ratkaisee yhtäön tai systeemin

fsolve - ratkaisee numeerisesti

```
> (x^3+1)/(x^2-x+1);simplify(%);
      x3 + 1
      -----
      x2 - x + 1
      x + 1
(12.1.1)
```

```
> expand((x^2-4)*(x+1)*(x-2)*(x^2+x+1));
x6 - 6x4 - 3x3 + 6x2 + 12x + 8
(12.1.2)
```

```
> factor(%);
(x + 2) (x + 1) (x2 + x + 1) (x - 2)2
(12.1.3)
```

Yhtälöistä

[SOL] s. 11-12

[HAM] s. Luku 6 s. 133 - 142

```
> yhtalot:={2*x-5*y=12, 12*x+4*y=17}; # Kyseessä on joukko
      yhtalot := {2 x - 5 y = 12, 12 x + 4 y = 17} (12.1.4)
```

```
> ratk:= solve(yhtalot, {x,y});
      ratk := { x = 133/68, y = -55/34 } (12.1.5)
```

Maple ei sijoita ratkaisuja muuttujille arvoiksi, vaan palauttaa sijoitussäännöt. Ne voidaan antaa suoraan

> #subs-komennolle. Mieti seuraavan komentojonon logiikkaa!

```
> subs(ratk, x), subs(ratk, y);
      133/68, -55/34 (12.1.6)
```

```
> X:=subs(ratk, x); Y:=subs(ratk, y);
      X := 133/68
      Y := -55/34 (12.1.7)
```

Tarkistaminen käy vaivattomasti:

```
> subs(ratk, yhtalot);
      {12 = 12, 17 = 17} (12.1.8)
```

Voidaan myös käyttää #lhs (left hand side) ja rhs (right hand side)-tapaa, mutta se on hiukan vähemmän elegantti ja myös altis Maplen harrastamalle järjestyksen vaihtumiselle ratkaisujoukossa. Niinpä tätä tyyliä ei voida ajaa automaattisesti, vaan tuloksiin on puututtava interaktiivisesti, mikä voi olla kiusallista, jos on "vakavahenkisistä" dokumentista kyse. Joka tapauksessa tämäkin tyyli kannattaa omaksua.

```
> rhs(ratk[1]); lhs(ratk[1]);
      133/68
      x (12.1.9)
```

```
> lhs(ratk[2])=rhs(ratk[2]);
      y = -55/34 (12.1.10)
```

Ratkaisun sijoittaminen muuttujan arvoksi

Tässä nyt on toisin sanoin selitetty samaa, pyyhi omasta dokustasi pois tai järjestä paremmin.

Tietysti näin:

```
> X:=rhs(ratk[1]); Y:=rhs(ratk[2]);
      X := 133/68
      Y := -55/34 (12.1.11)
```

Turvallisempi ja elegantimpi tapa (jota edellä mainostettiin):

Suorita ratk-yhtälöiden ilmaisemat korvaamiset lausekkeessa (pelkkä) x ja suorita ratk-yhtälöiden ilmaisemat korvaamiset lausekkeessa (pelkkä) y .

Edellisessä y-yhtälö jää vaille käyttöä (kun pelkkä x ei sisällä y:tä) ja jälkimmäisellä vastaavasta syystä x-yhtälö. ratk-joukon järjestys ei näyttele mitään osaa.

```
> X:=subs(ratk, x); Y:=subs(ratk, y);
```

$$\begin{aligned} X &:= \frac{133}{68} \\ Y &:= -\frac{55}{34} \end{aligned} \quad (12.1.12)$$

Tehokkain, mutta turvattomin tapa, joskus kylläkin tosi kätevä: `assign(ratk);`

```
> assign(ratk);
> x;y;yhtalot;
```

$$\begin{aligned} &\frac{133}{68} \\ &-\frac{55}{34} \\ &\{12 = 12, 17 = 17\} \end{aligned} \quad (12.1.13)$$

```
> solve(yhtalot,{x,y}); # assign-komennon jälkeen x ja y eivät
ole vapaita muuttujia, josta syystä solve menee virheeseen.
```

Warning, solving for expressions other than names or functions is not recommended.

Error, (in solve) a constant is invalid as a variable, -55/34, 133/68

`assign(ratk);` toimii ikäänkuin ratk-yhtälöissä **yhtälömerkki (=)** vaihdettaisiin **sijoitusmerkkien (:=)**

Yhtälöiden toteutuminen saadaan hyvin kätevästi tarkistetuksi kirjoittamalla vain **yhtalot;**

Sensijaan yhtälöiden ratkaisua ei voida toistaa, **ennakuin muuttujat x ja y on vapautettu**

arvoistaan.

```
> x:='x':y:='y':solve(yhtalot,{x,y});
```

$$\left\{ x = \frac{133}{68}, y = -\frac{55}{34} \right\} \quad (12.1.14)$$

10. Piirtoa

?PlottingGuide

[SOL] ss. 10 - 11

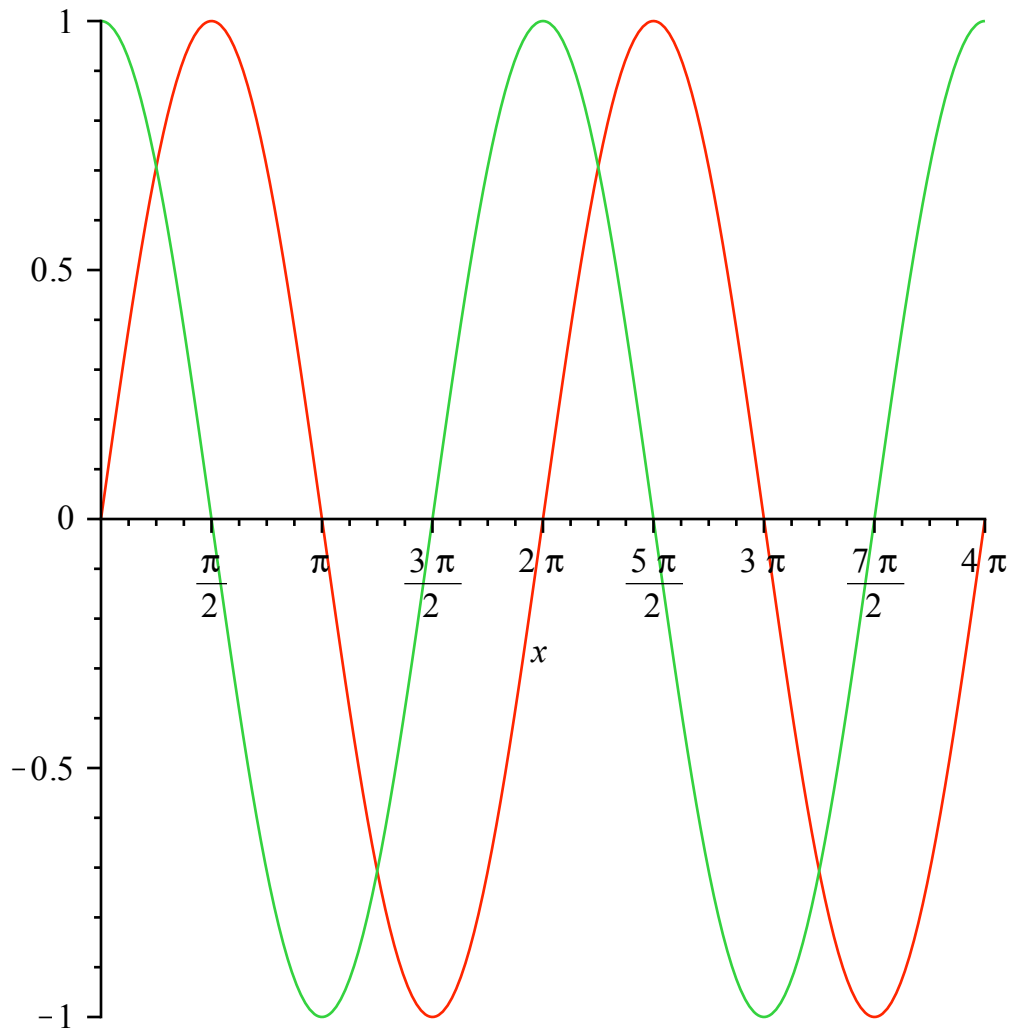
[HAM] Luku 3 ss. 89 - 100

Tavallinen käyrien piirto

Käyräparven voi sulkea joko joukkosulkuihin `{ }` tai listasulkuihin `[]`.

Jälkimmäinen lienee suositeltavaa, koska järjestys on silloin käyttäjän hallinnassa (esim jos halutaan väreit hallitusti, tms).

```
> plot([sin(x),cos(x)],x=0..4*Pi);
```



Ota hiiren vasemmalla kiinni kuvasta, saat kehyksen ja uudet valikot.

Valitse sama mittakaava akseleilla (1:1)

Jos grafiikka alkaa hidastaa työarkin selausta, kannattaa valita EDIT-valikon alimmasta, Remove output. Sitä kannattaa käyttää muutenkin silloin tällöin. Työarkki tiivistyy (ja nopeutuu) kummasti. Outputit saa takaisin saman EDIT-valikon toiseksi alimmaisesta execute worksheet

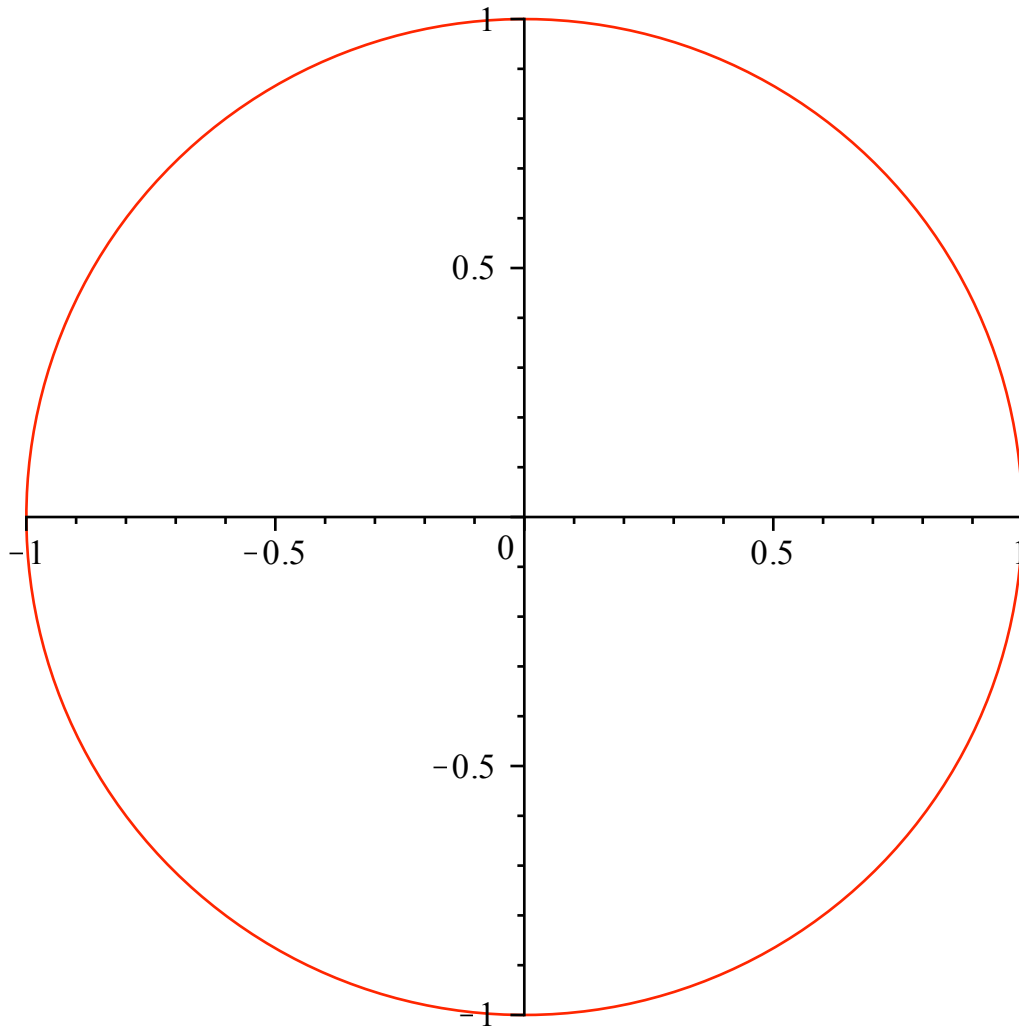
Huomaa: Pi on Iso P, pieni i. (pi kirjoittuu oikean näköisesti, mutta Maple ei tunnista sitä muuksi kuin kreikkalaiseksi kirjaimeksi.) [No johan tuosta huomauteltiin.]

Parametrimuotoinen piirto, `matrixplot(logtaulu);`

Syntaksi poikkeaa aika vähän, tämä täytyy vain oppia (tai katsoa aina uudelleen helpistä).

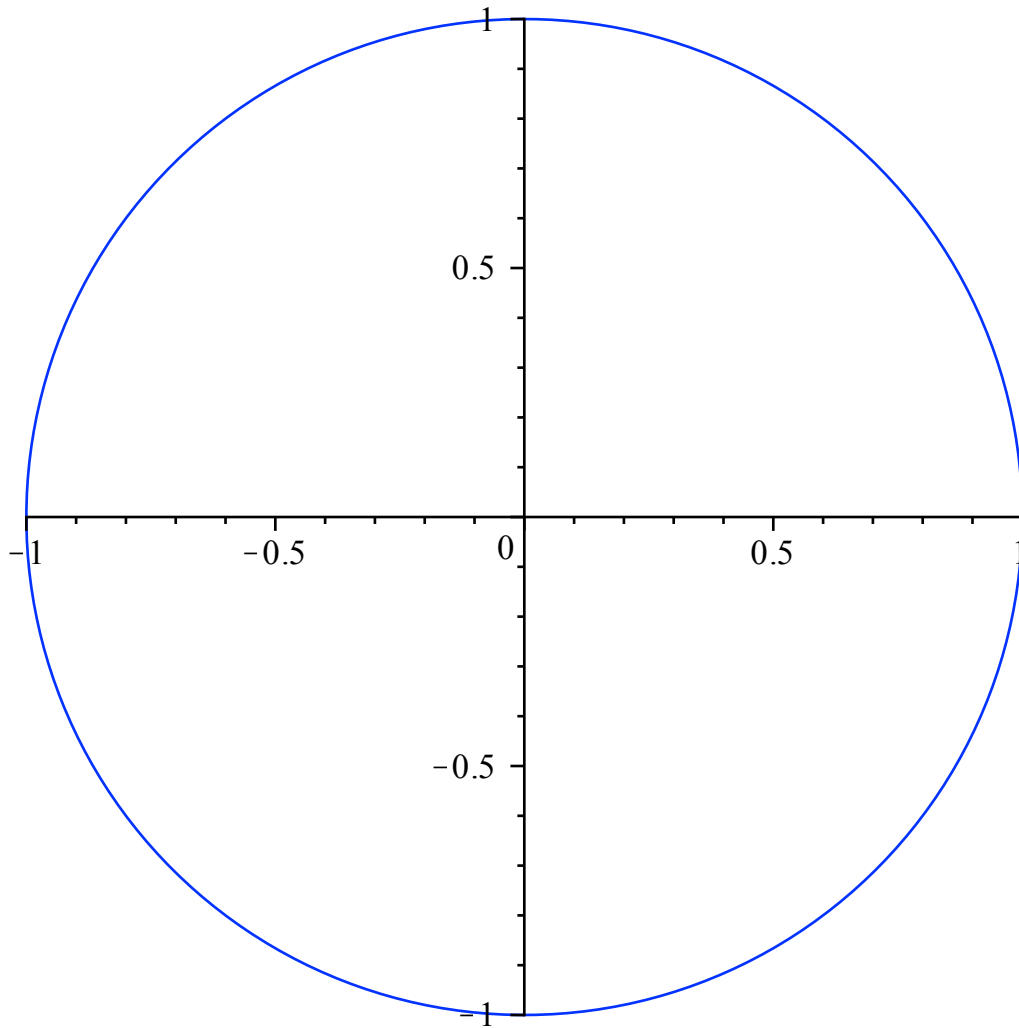
Aina ei ole hauskaa joutua valitsemaan hirellä, samaskaalaisuus (ja monia muita) voidaan antaa plot-komennon tarkentimena.

```
> plot([cos(t), sin(t), t=0..2*Pi], scaling=constrained);
```



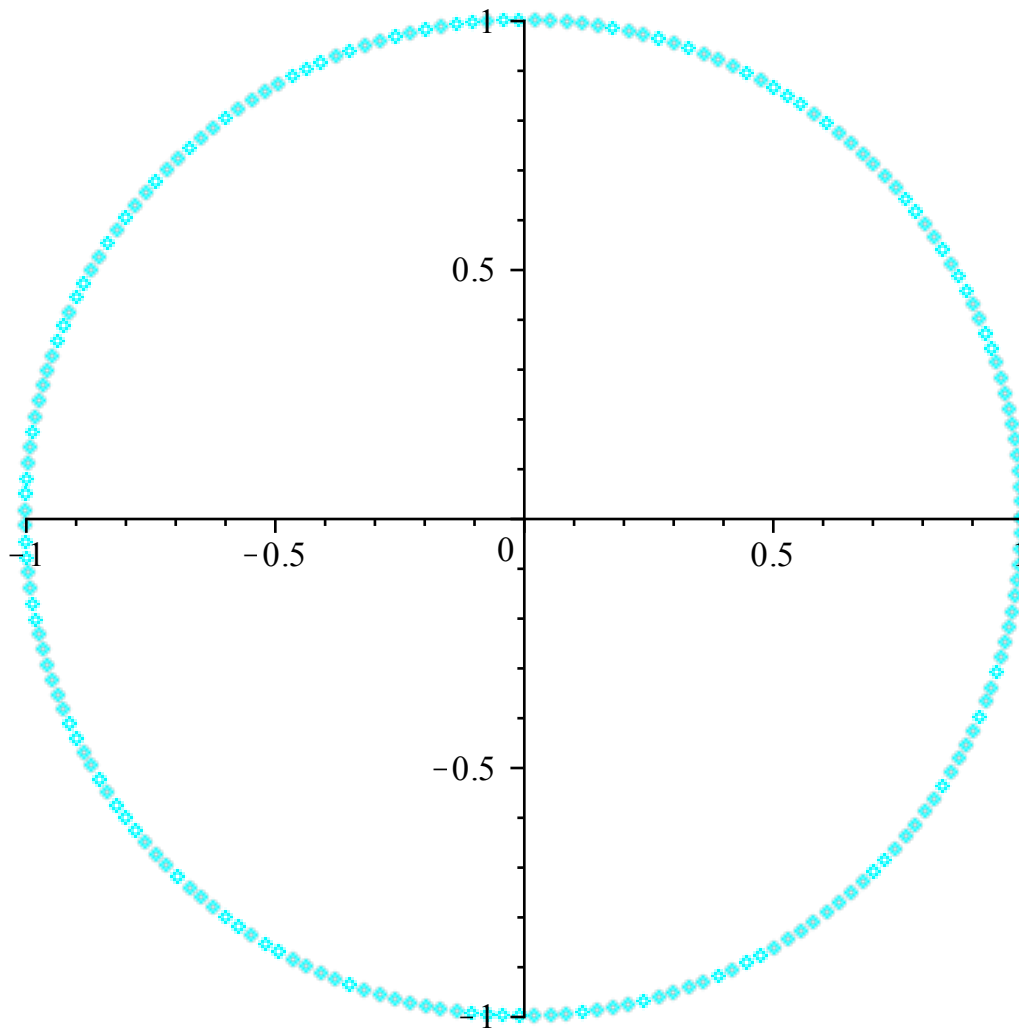
Pisteet voidaan yhtä hyvin ajatella kompleksitason pisteinä. `complexplot` osaa käsitellä suoraan reaali- ja imaginaariosien kompleksiarvoista funktiota. Tässä on sekin mukava piirre, että syntaksi on luonnollisempi ja siten helpompi muistaa kuin tuossa parametrimuotoisessa R^2 -piirroksessa.

```
> with(plots):  
> complexplot(cos(t)+I*sin(t), t=0..2*Pi, scaling=constrained,  
color=blue);
```



Edellinen voidaan ilmaista lyhyemmin eksponenttifunktion (Eulerin kaavan) avulla. Piirretään vaihteeksi pelkät pisteet.

```
> complexplot(exp(I*t), t=0..2*Pi, scaling=constrained, color=cyan,  
style=point);
```



▼ Taulukoidun datan piirto ("listaplotti")

Sometimes we will be interested in functions defined in terms of a discrete table of values rather than a formula. For example, consider the following table of temperatures recorded at various times on a spring day in Raleigh.

Time:	6:00 a.m.	10:00 a.m.	12:00 p.m.	4:00 p.m.	5:00 p.m.

Temp:	45 deg.	57 deg.	65 deg.	67 deg.	66 deg.

Table of Temperatures on a Spring Day in Degrees Fahrenheit

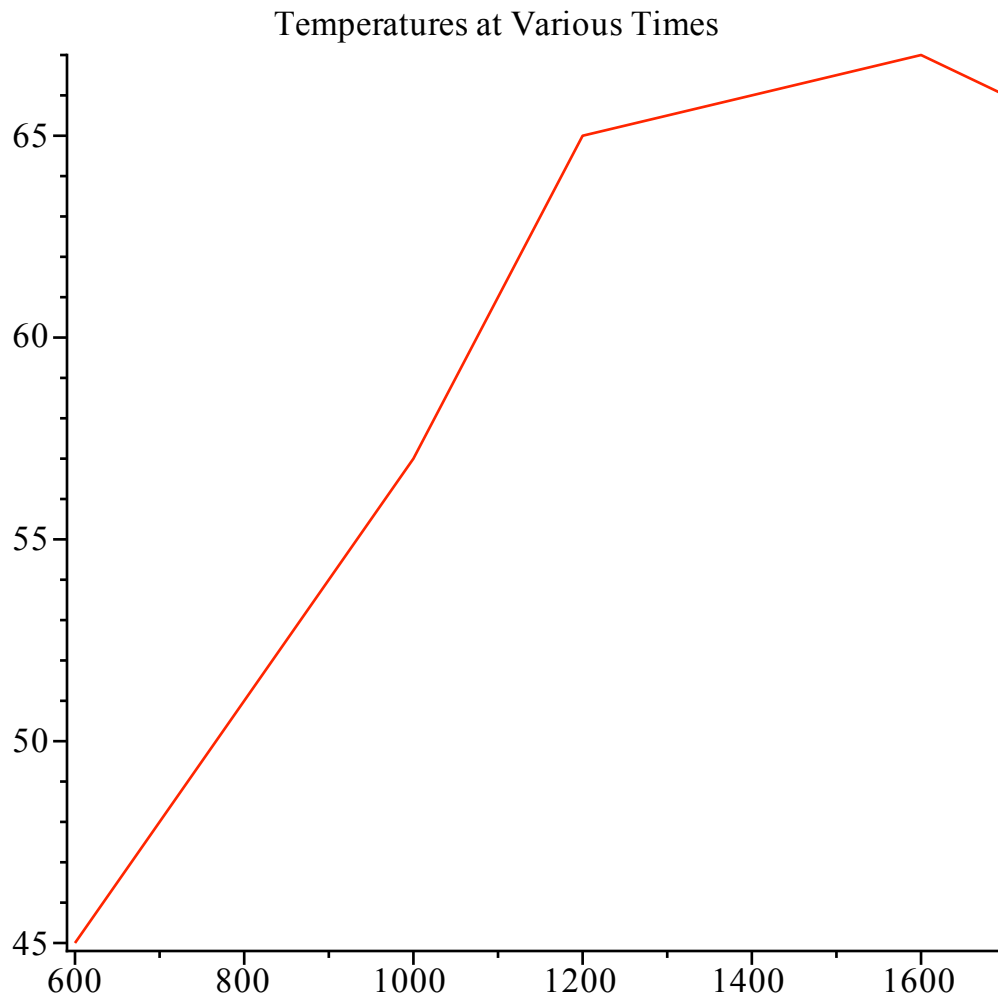
To plot this data you must first define the points as a list. Since the time is recorded in a cyclic fashion, we will plot the times in the so-called "military style", i.e. 6:00 a.m. is 0600 and 6:00 p.m. is 1800.

```
> ListOfPoints := [[600,45], [1000,57], [1200,65], [1600,67],
  [1700,66]];
  ListOfPoints := [[600, 45], [1000, 57], [1200, 65], [1600, 67], [1700, 66]] (13.3.1)
```

The **plot** command allows you to plot a list of points and to connect them with straight lines

(unless you choose another option). This is illustrated in the next command. Notice also the option which produces a title on the plot.

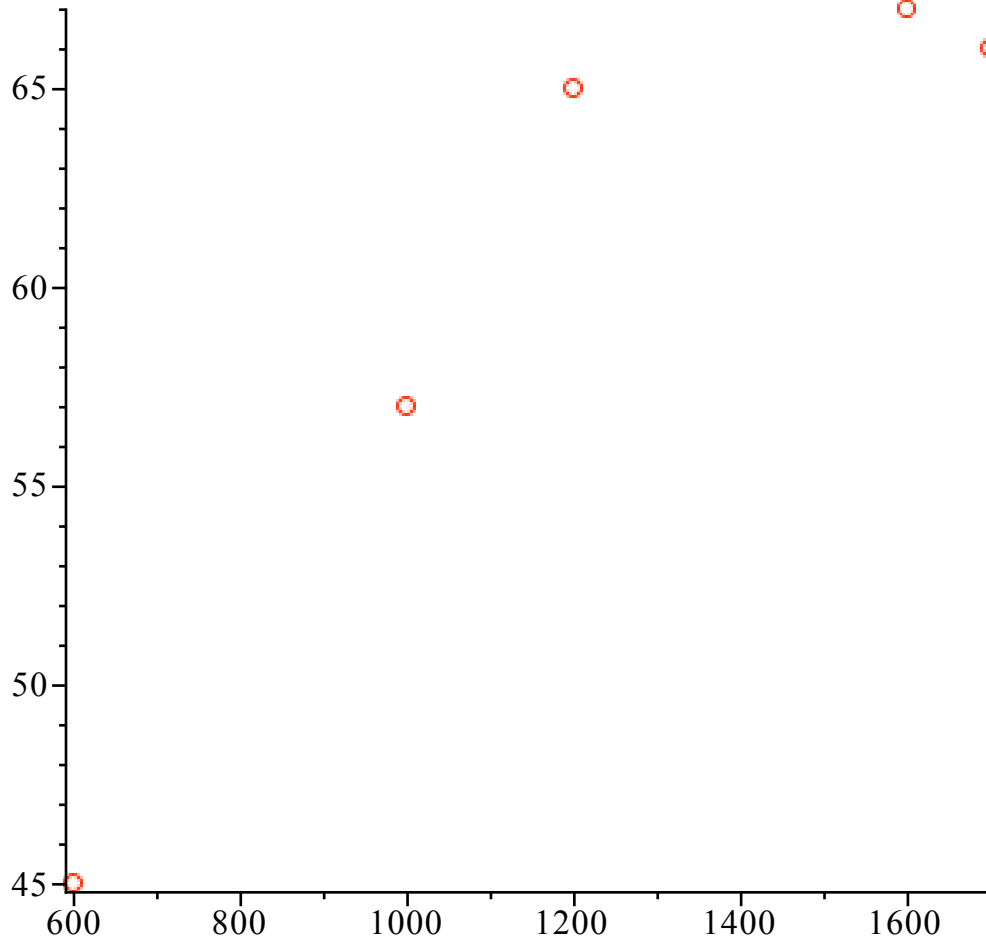
```
> plot(ListOfPoints, title="Temperatures at Various Times");
```



While the data can be plotted in this way, because the data is given for a discrete set of points you may prefer the **style=POINT** option as illustrated in the next command. Note also the effect of the **symbol=CIRCLE** option.

```
> plot(ListOfPoints, style=POINT, symbol=CIRCLE, symbolsize=15,  
title=`Temperatures at Various Times`);
```

Temperatures at Various Times



Exercise

1. Plot the graph of $f(x) = e^{-x} \sin(2x)$ over the interval $[-\pi, \pi]$.
2. Make a plot of the points in the following table:

X:	-1	0	1	2	3
Y:	2.72	1	0.368	0.135	0.0498

Make two plots, one with the points connected and the other with only the data points.

Miten saadaan kätevästi koordinaattiparien lista $[[x_1, y_1], [x_2, t_2], \dots]$?

Olkoon annettu numeeriset listat (Matlabissa sanoisimme vektorit) X ja Y. Matlabin plot toimii tyyliin `plot(x,y)`;

Maplessa täytyy muodostaa parien lista vaikkapa tähän tapaan:

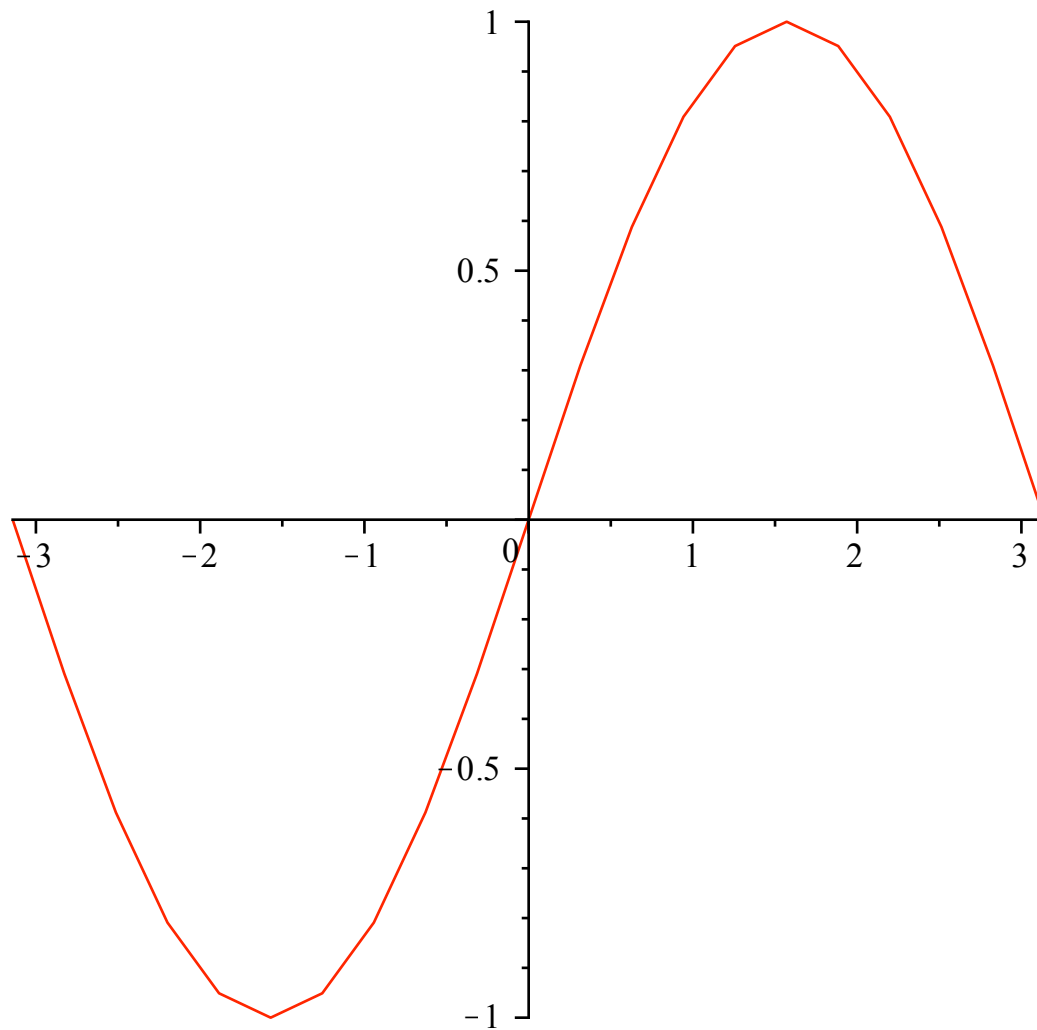
```
> a:=-Pi:b:=Pi:N:=20:h:=(b-a)/N:x:=seq(evalf(a+i*h),i=0..N);  
x := -3.141592654, -2.827433389, -2.513274123, -2.199114858, -1.884955592, (13.3.2)  
-1.570796327, -1.256637062, -0.9424777962, -0.6283185308, -0.3141592654,
```

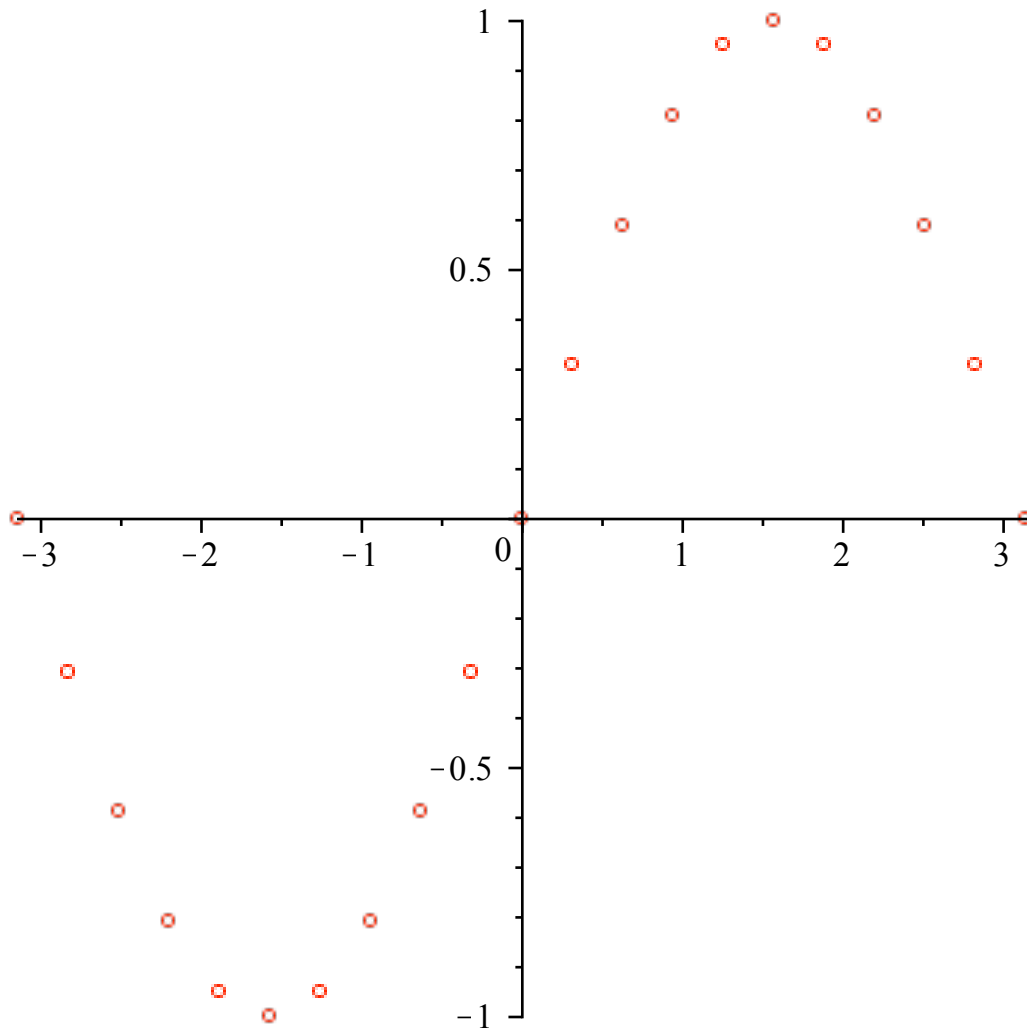
0., 0.3141592654, 0.6283185308, 0.9424777962, 1.256637062, 1.570796327,
1.884955592, 2.199114858, 2.513274123, 2.827433389, 3.141592654

Olkoot y-arvot vaikkapa sin-funktion arvoja x-pisteissä. Parijono saataisiin nyt näin:

```
> xy:=[seq([x[i],sin(x[i])],i=1..N+1)];  
xy := [[ -3.141592654, 4.102067615 10-10 ], [ -2.827433389, -0.3090169936 ], [  
-2.513274123, -0.5877852522 ], [ -2.199114858, -0.8090169941 ], [  
-1.884955592, -0.9510565163 ], [ -1.570796327, -1. ], [ -1.256637062,  
-0.9510565165 ], [ -0.9424777962, -0.8090169944 ], [ -0.6283185308,  
-0.5877852524 ], [ -0.3141592654, -0.3090169944 ], [ 0., 0. ], [ 0.3141592654,  
0.3090169944 ], [ 0.6283185308, 0.5877852524 ], [ 0.9424777962, 0.8090169944 ],  
[ 1.256637062, 0.9510565165 ], [ 1.570796327, 1. ], [ 1.884955592, 0.9510565163 ],  
[ 2.199114858, 0.8090169941 ], [ 2.513274123, 0.5877852522 ], [ 2.827433389,  
0.3090169936 ], [ 3.141592654, -4.102067615 10-10 ] ]  
> plot(xy);plot(xy,style=point,symbol=circle);
```

(13.3.3)





Huom: `plots`-pakkauksessa on `listplot` ja `pointplot`. Maplelle luonteenomaista on, että siinä on joukko redundanteja funktioita. Mielestäni on parempi oppia käyttämään harvempaa ydinfunktiojoukkoa kuin omaksua monenlaisia synonyymejä. Niissä voi toki olla joitakin uusia mahdollisuuksia, mutta esim. näissä ei välttämättä ole (kuka noita kaikkia tuhansia ehtii penkoa).

Taulukointi

Tässä teimme itse asiassa taulukon, joka on havainnollisempi usein matriisina.

```
> txy:=matrix(xy):linalg[transpose](xy):
```

Tilan säästämiseksi otetaan vähemmän dataa esimerkkiimme. Jätetäänpä myös `evalf` pois.

```
> a:=-Pi:b:=Pi:N:=8:h:=(b-a)/N:x:=seq(a+i*h,i=0..N);
xy:=[seq([x[i],sin(x[i])],i=1..N+1)];
matrix(xy);
```

$$x := -\pi, -\frac{3}{4}\pi, -\frac{1}{2}\pi, -\frac{1}{4}\pi, 0, \frac{1}{4}\pi, \frac{1}{2}\pi, \frac{3}{4}\pi, \pi$$

$$xy := \left[[-\pi, 0], \left[-\frac{3}{4}\pi, -\frac{1}{2}\sqrt{2} \right], \left[-\frac{1}{2}\pi, -1 \right], \left[-\frac{1}{4}\pi, -\frac{1}{2}\sqrt{2} \right], [0, 0], \left[\frac{1}{4}\pi, \frac{1}{2}\sqrt{2} \right], \left[\frac{1}{2}\pi, 1 \right], \left[\frac{3}{4}\pi, \frac{1}{2}\sqrt{2} \right], [\pi, 0] \right]$$

$$\begin{bmatrix} -\pi & 0 \\ -\frac{3}{4}\pi & -\frac{1}{2}\sqrt{2} \\ -\frac{1}{2}\pi & -1 \\ -\frac{1}{4}\pi & -\frac{1}{2}\sqrt{2} \\ 0 & 0 \\ \frac{1}{4}\pi & \frac{1}{2}\sqrt{2} \\ \frac{1}{2}\pi & 1 \\ \frac{3}{4}\pi & \frac{1}{2}\sqrt{2} \\ \pi & 0 \end{bmatrix}$$

(13.3.4)

> `linalg[transpose](xy);`

$$\begin{bmatrix} -\pi & -\frac{3}{4}\pi & -\frac{1}{2}\pi & -\frac{1}{4}\pi & 0 & \frac{1}{4}\pi & \frac{1}{2}\pi & \frac{3}{4}\pi & \pi \\ 0 & -\frac{1}{2}\sqrt{2} & -1 & -\frac{1}{2}\sqrt{2} & 0 & \frac{1}{2}\sqrt{2} & 1 & \frac{1}{2}\sqrt{2} & 0 \end{bmatrix}$$

(13.3.5)

> `array(xy);` # Taas synonyymi, mutta tiettyjä erojakin yleisyydessä / matriisilaskukyvyissä.

(13.3.6)

$$\begin{bmatrix} -\pi & 0 \\ -\frac{3}{4}\pi & -\frac{1}{2}\sqrt{2} \\ -\frac{1}{2}\pi & -1 \\ -\frac{1}{4}\pi & -\frac{1}{2}\sqrt{2} \\ 0 & 0 \\ \frac{1}{4}\pi & \frac{1}{2}\sqrt{2} \\ \frac{1}{2}\pi & 1 \\ \frac{3}{4}\pi & \frac{1}{2}\sqrt{2} \\ \pi & 0 \end{bmatrix}$$

(13.3.6)

▼ Grafiikoiden yhdistäminen, plots[display] ja textplot

Tässä tarvitaan lisägrafiikkapakkaus **plots**

```
> restart with (plots):
```

Selvitä itsellesi, mitä seuraavassa tehdään. Mieti tarkkaan, miksi jossain pitää antaa plot:lle argumentiksi

$f(x)$ ja miksi taas jossain esim. tang. Voit sitten huvitella vaihtelemalla funktion määritelmää ja / tai pistettä x_0 . Kirjoitetaan pieni malliskripti, jonkalaisia voit tehdä moninaisissa yhteyksissä.

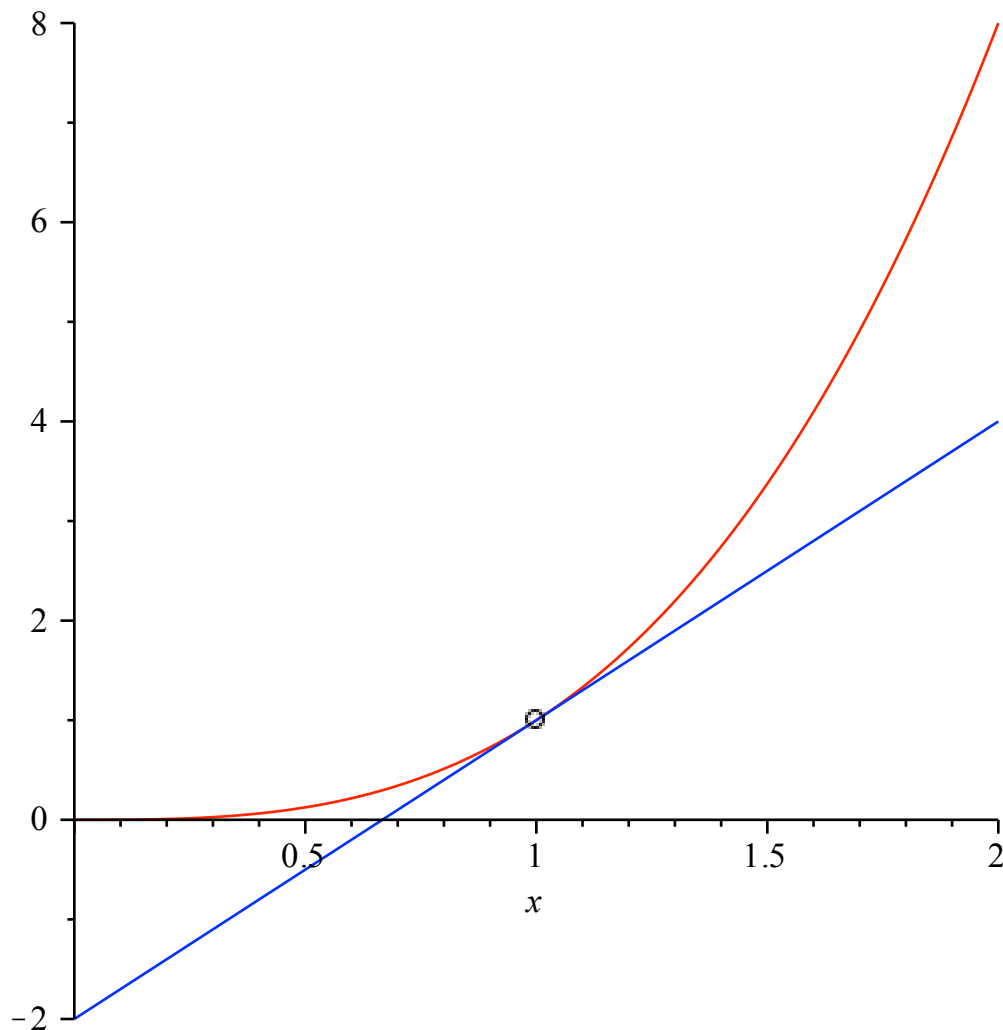
```
> x0:=1: f:=x->x^3: # Vaihtuva syöte
> df:=diff(f(x),x): kk:=subs(x=1,df);
kk:=3
```

(13.4.1)

```
> x0:=1: tang:=f(x0)+kk*(x-x0);
tang := -2 + 3 x
```

(13.4.2)

```
> fkuva:=plot(f(x),x=0..2,color=red): tangkuva:=plot(tang,x=0.
.2,color=blue): p0kuva:=plot([[x0,f(x0)]],style=point,symbol=
circle,symbolsize=15,color=black):
> display([fkuva,tangkuva,p0kuva]);
```



Kokeile (mutta vain kerran elämässäsi, silloinkin syvästi katuen, että olit yllytyshullu), mitä tapahtuu, kun vaihdat $(:)$ \rightarrow $(;)$ vaikkapa **fkuva:=; ...** yllä. PLOT-tietorakenne paljastuu.

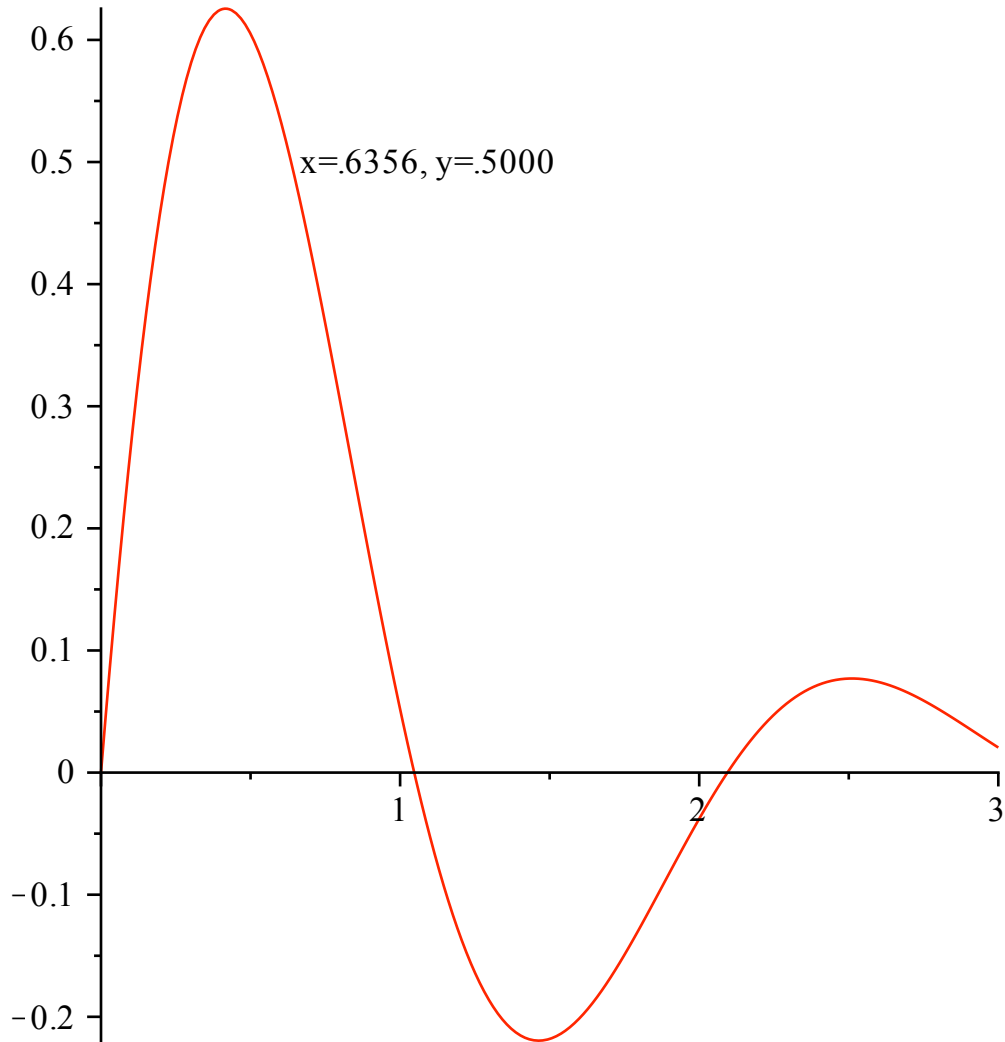
- No, mene edit-valikkoon ja \rightarrow remove output, kyllä se siitä.

Sometimes it will be desirable to **print text on a plot**. The **plots** procedure **textplot** will allow this. The following commands will serve as examples of the **textplot** and **display** routines. Make sure you notice where colons and semicolons are used in these commands.

In the next commands it is important to use colons to punctuate the first two statements. Otherwise Maple will output an entire page of text describing the plot structure rather than the graph. The **textplot** in the second command causes the text $x=0.6356, y=0.5000$ to be printed at the point $(.6356, .5000)$. The statement **align=RIGHT** aligns this text to the right of the point.

```
> f := x -> exp(-x)*sin(3*x);
plot1 := plot(f, 0..3):# tai plot1:=plot(f(x),x=0..3):
plot2 := plots[textplot]([0.6356, 0.5000, "x=.6356, y=.5000"],
align=RIGHT):
plots[display]({plot1, plot2});
```

$$f := x \rightarrow e^{-x} \sin(3x)$$



Huom:

plots-pakkauksen (kuten muidenkin pakkausten) funktioita voidaan käyttää myös lataamatta koko pakkausta, tällöin pakkauksen nimeä ikäänkuin indeksoidaan ao. funktion nimellä tyyliin `plots[display]`

3d-grafiikkaa, animaatiot

Otetaan lämmönjohtumisesimerkki, jossa valaistaan

3d-pinnan piirtoa

Animaatiota

3d-kuvan projektiokäyräparven piirtoa

Tarkastellaan sivuiltaan lämpöeristettyä sauvaa, jonka päät upotetaan hetkellä $t=0$ jäävesisäiliöihin (0 astetta) ja jonka alkulämpötilajakauma on

```
> f:=x->100*sin(Pi*x/80);
```


$$f := x \rightarrow 100 \sin\left(\frac{1}{80} \pi x\right) \quad (13.5.1)$$

Huomaa funktiomäärittäminen.

Olkoon sauvan pituus $L=80$.

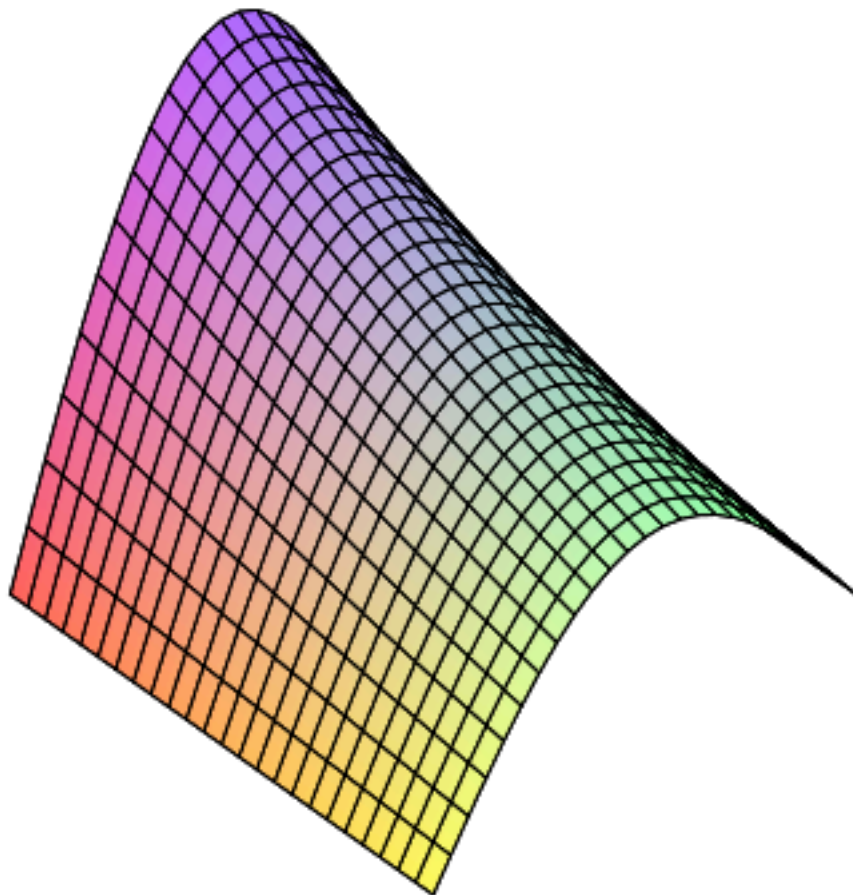
Tässä tapauksessa lämpöyhtälön ratkaisuna olevasta Fourier-sarjasta tulee vain yksi termi. Ratkaisu on (sopivalla lämmönjohtumiskertoimella)

```
> u := (x, t) -> 100 * sin((Pi * x) / 80) * exp(-0.001785 * t);
```

$$u := (x, t) \rightarrow 100 \sin\left(\frac{1}{80} \pi x\right) e^{-0.001785 t} \quad (13.5.2)$$

Pintapiirros

```
> with(plots):  
> plot3d(u(x, t), x=0..80, t=0..400);
```



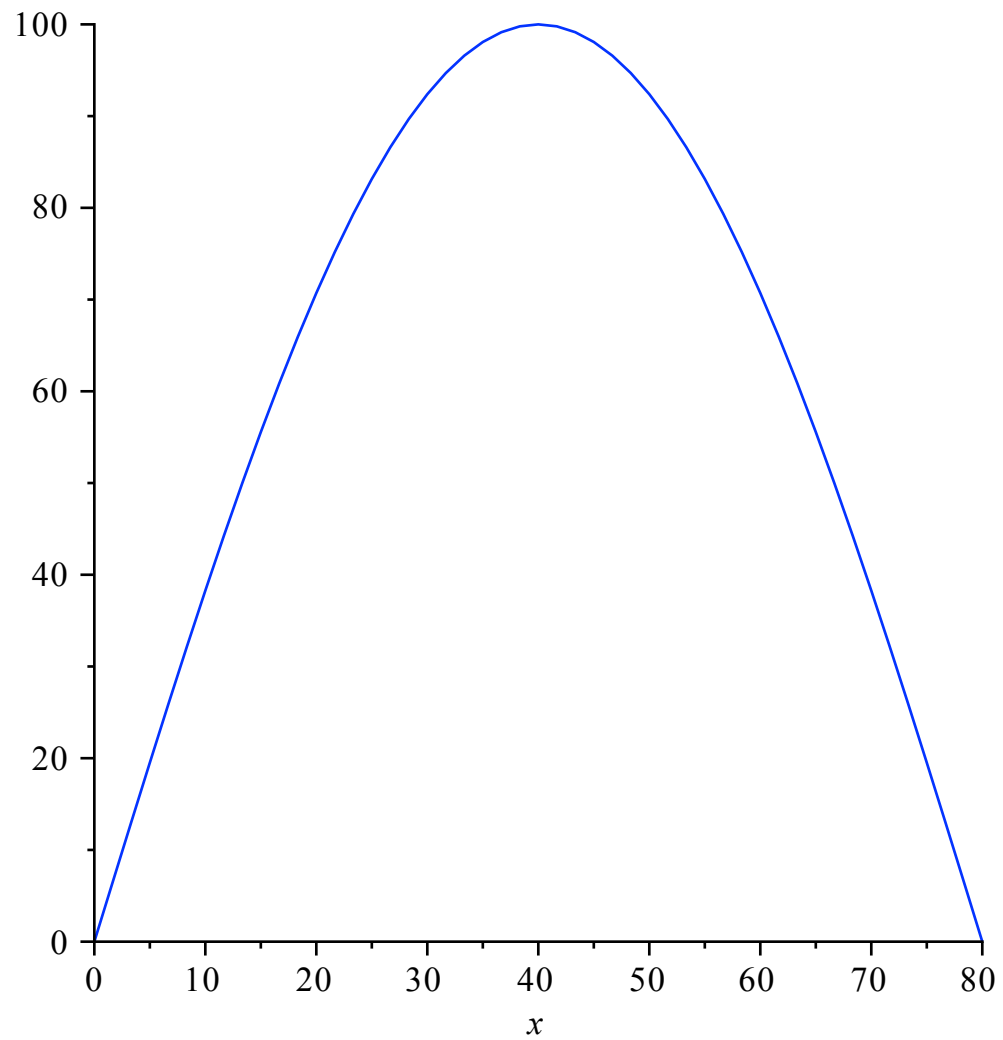
Klikkaa hiirellä kuvaan (ja tarvittaessa valitse "boxed") työkalunauhasta.

Kierrä kuvaa hiirellä.

Kokeile STYLE-valikosta PATCH ja PATCH with contour valintoja.

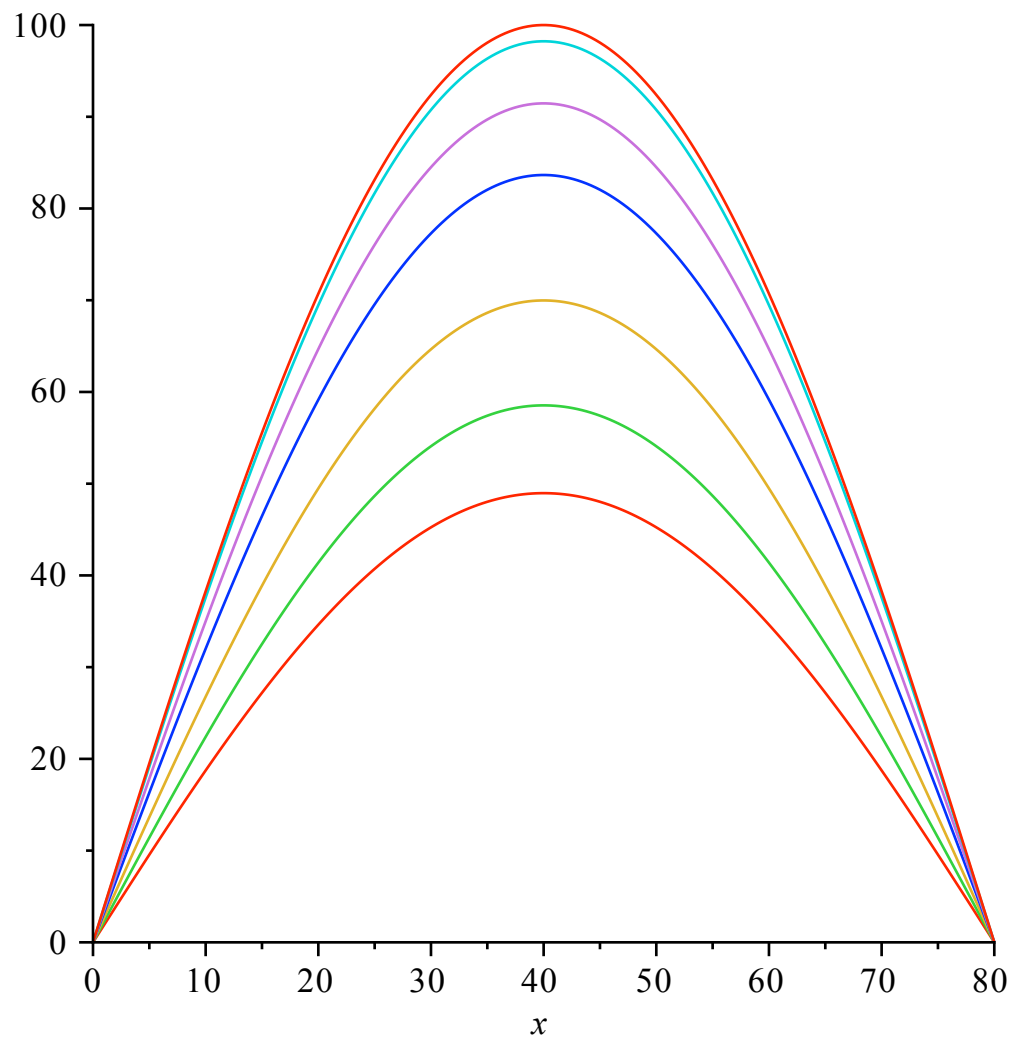
Animaatio

```
> animate(u(x,t), x=0..80, t=0..300, frames=30, color=blue);
```

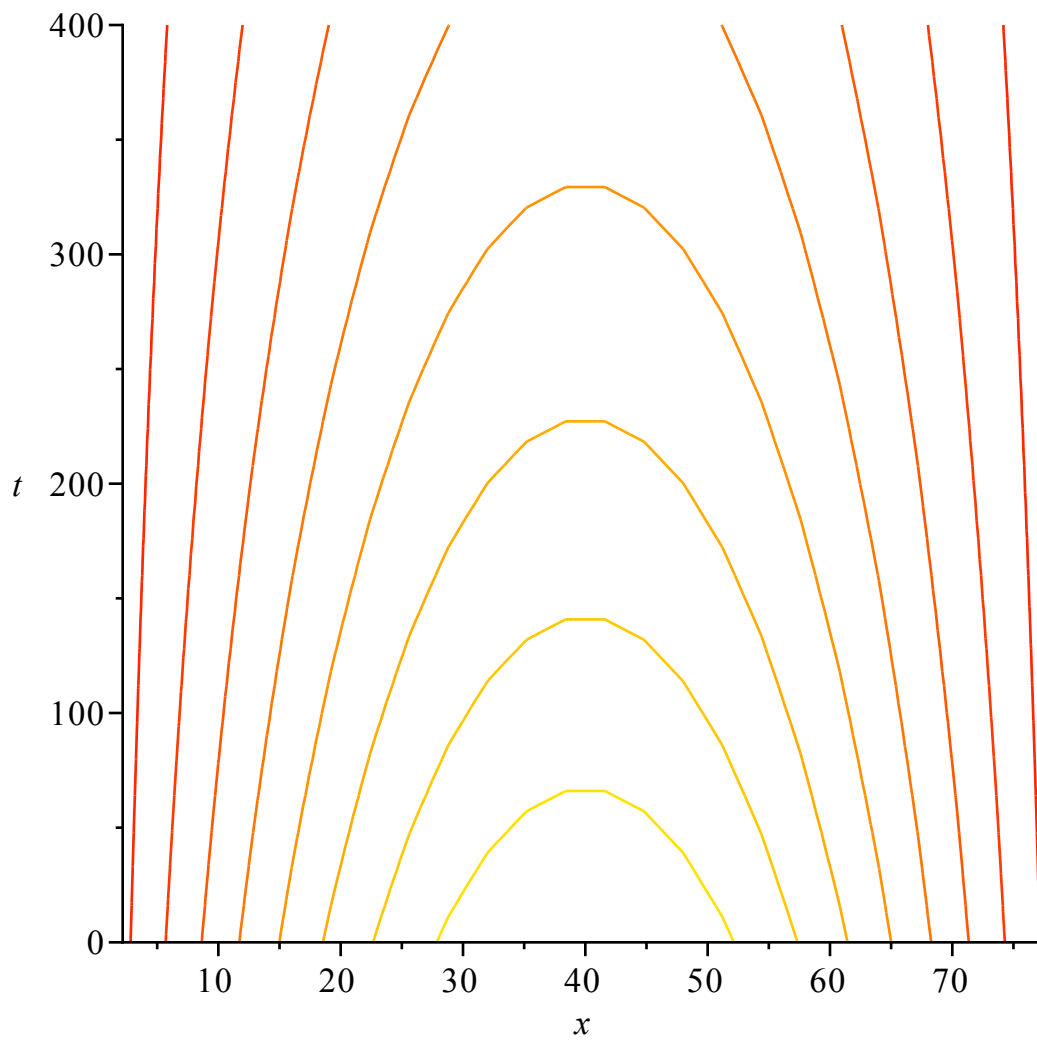


Lämpötilaprofiilit ja korkeuskäyrät

```
> plot({seq(u(x,t), t=[0,10,50,100,200,300,400])}, x=0..80);
```



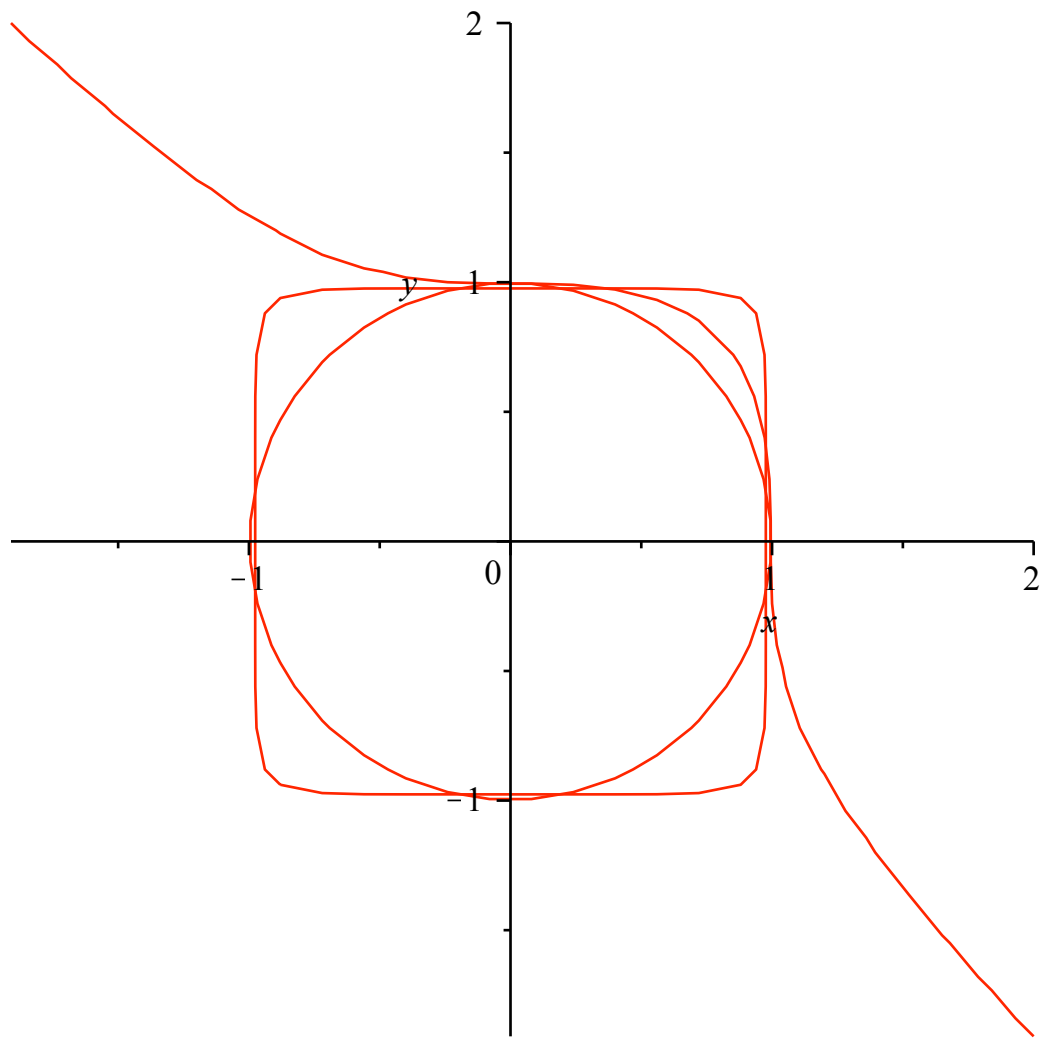
```
> contourplot(u(x,t),x=0..80,t=0..400);
```



▼ **Implicitplot**

Tämä on periaatteessa sama kuin yhden korkeuskäyrän piirto.

```
> implicitplot({x^2+y^2=1,x^3+y^3=1,x^10+y^10=1},x=-2..2,y=-2..2);
```



Tuohon tekisi mieli laittaa loppuun esim: `,color=[red,blue,black]);`
mutta näköjään kaikki **plot**:n hyväksymät optiot eivät tässä toimi.