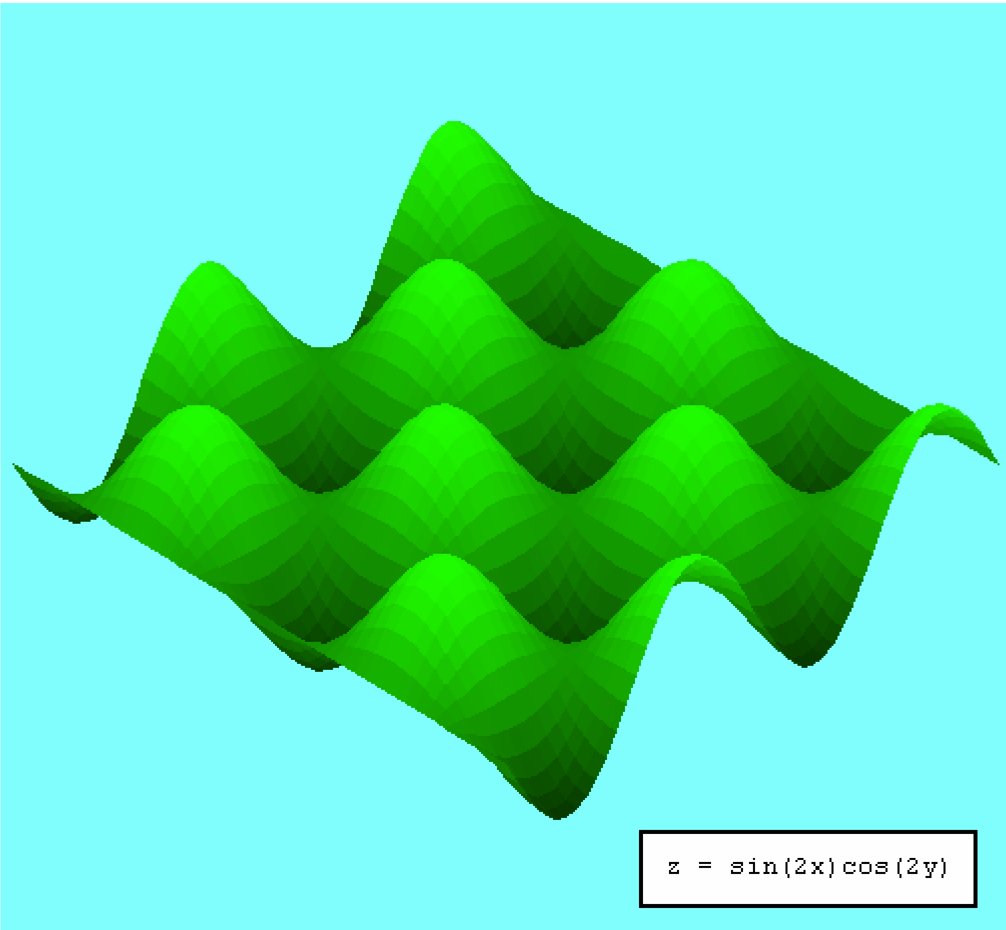
Winplot

Dette dokumentet er en veiledning i bruk av graftegningsprogrammet *Winplot*. Det er satt sammen av elleve enkeltveiledninger som opprinnelig ble skrevet i forbindelse med fjernundervisningskurset "Matematikk 2", høsten 2003 og våren 2004.



Innhold

Generelt

Den versjonen av *Winplot* som beskrives i dette dokumentet, er den du finner i samme mappe som denne veiledningen.

Winplot oppdateres kontinuerlig, og du kan hente siste versjon her:
http://math.exeter.edu/rparris/winplot.html
Men denne versjonen vil sannsynligvis avvike fra beskrivelsen i dette dokumentet.

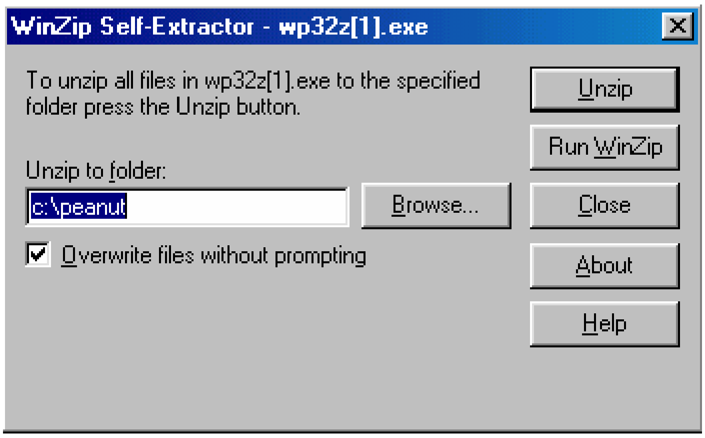
En engelsk veiledning kan du finne her:
http://matcmadison.edu/alehnen/winptut/winpltut.htm

Winplot - Introduksjon

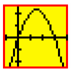
Dette avsnittet gir en første innføring i bruk av graftegningsprogrammet *Winplot*

Installere *Winplot*

Hent *Winplot* fra en av linkene nevnt under "Generelt". Velg å kjøre programmet fra gjeldende plassering/åpne programmet når du får spørsmål om dette. Når programmet er klart til installasjon, kommer det opp en dialogboks som ser slik ut:



Klikk på "Unzip", deretter på "Close". I mappa "c:\peanut" vil du nå finne *Winplot*-ikonet. Det ser slik ut:

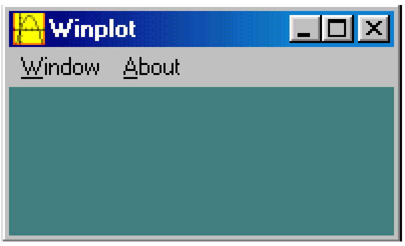


Så er det bare å klikke på ikonet, og *Winplot* starter.

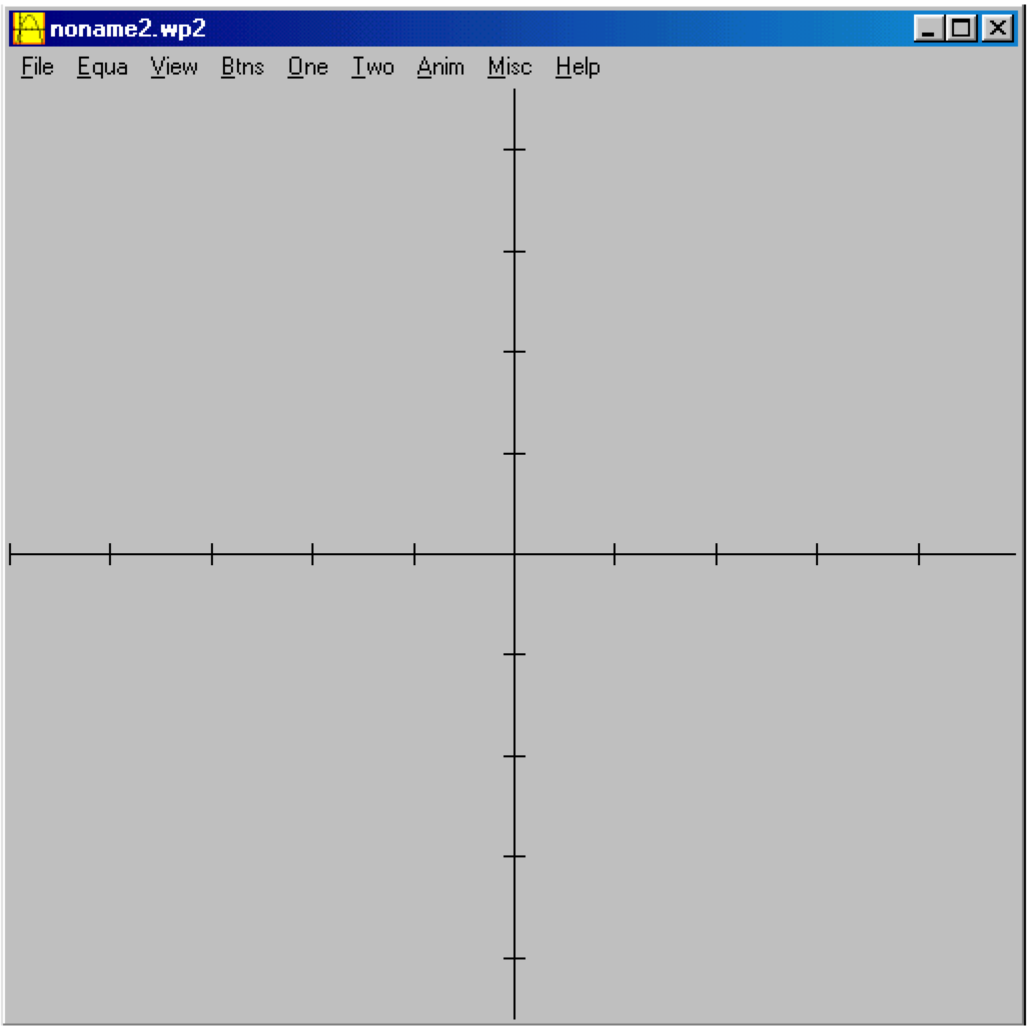
Lag en snarvei til skrivebordet hvis du vil ha *Winplot* lett tilgjengelig.

Lage et plott

Når *Winplot* starter, kommer vinduet under opp. Det vil heretter bli referert til som "hovedvinduet".



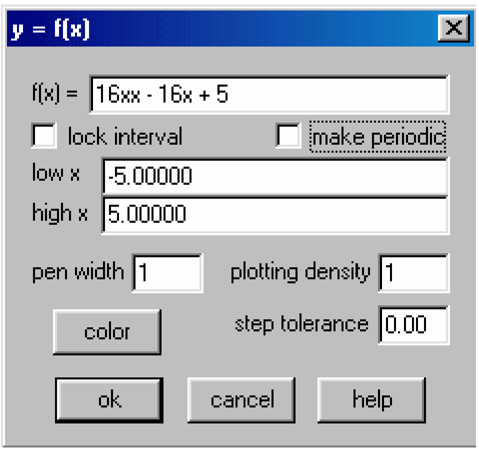
Vi skal i første omgang bare beskjeftige oss med todimensjonale plott. Vi velger "Window" - "2-dim", og vinduet under kommer opp. Det vil heretter bli referert til som "plottevinduet".



Vi skal etter hvert vise hvordan en endrer utseendet på dette vinduet. *Winplot* husker endringene, slik at en ikke behøver gjøre det om igjen for hver gang *Winplot* startes.

Som et introduksjonseksempel skal vi lage et plott av funksjonen $y = 16x^2 - 16x + 5$.

Velg "Equa" - "Explicit…". Dette gir oss muligheten til å plotte *y* som en funksjon av *x*. Vi får opp en dialogboks der vi fyller ut "f(x)=" med "16xx - 16x + 5", som vist under:



Når du klikker "ok", tegnes kurven i plottevinduet, og det kommer opp en boks som heter "inventory for window".

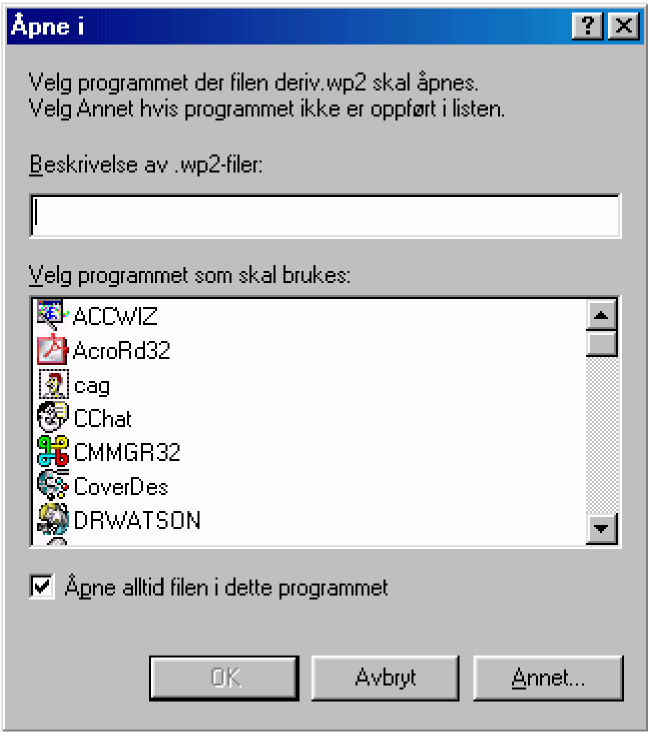
Lagre og åpne et plott

Du lagrer et plott ved å velge "File" - Save As" i plottevinduet. Plottevinduet kan deretter lukkes ved å trykke på X-knappen øverst i høyre hjørne.

Hvis du i hovedvinduet har merket av "Window" - "Open last", vil plottet du arbeidet med sist, automatisk komme opp neste gang plottevinduet åpnes.

For å åpne et plott du har fra før, velger du "File" - "Open…" i plottevinduet.

For å kunne åpne ei *Winplot*-fil ved å dobbeltklikke på den, må en ty til et lite triks: Dobbeltklikk på ei *Winplot*-fil. En dialogboks kommer opp. I *Windows 98* er den som vist under, i andre versjoner av *Windows* kan den se annerledes ut:



Ikke velg blant de programmene som er listet opp. Velg "Annet", klikk deg fram til "C:\peanuts\winplot.exe", og klikk så "OK".

Symboler og funksjoner

I eksemplet i forrige avsnitt, skrev vi x^2 som *xx*. Men en kunne også ha skrevet *x^2* eller *x*x*.

De vanlige regnesymbolene som brukes i *Winplot* er:

- Multiplikasjon: * (I tilfeller der det ikke er tvetydighet, kan * sløyfes)
- Divisjon: /
- Addisjon: +
- Subtraksjon: -
- Potenser: ^

Har du flere ledd over/under en brøkstrek, må du bruke parentes. $\frac{x+2}{x-3}$ skrives for eksempel slik: (x+2)/(x-3).

Winplot har en mengde innebygde funksjoner. Her er en liste over en del av dem.

- abs(x) $|x|$
- sgn(x) Fortegnet til *x*, dvs. 1 hvis $x \geq 0$, og -1 hvis $x < 0$.
- sqr(x) $\sqrt{x}$
- root(n, x) $\sqrt[n]{x}$
- exp(x) e^x
- ln(x) Den naturlige logaritmen til *x*.
- log(x) Logaritmen til *x* med 10 som grunntall.
- sin(x) sinus til en vinkel *x*, oppgitt i radianer.
- cos(x) cosinus til en vinkel *x*, oppgitt i radianer.
- tan(x) tangens til en vinkel *x*, oppgitt i radianer.
- arcsin(x) Den omvendte sinusfunksjonen, $\sin^{-1} x$.
- arccos(x) Den omvendte cosinusfunksjonen, $\cos^{-1} x$.
- arctan(x) Den omvendte tangensfunksjonen, $\tan^{-1} x$.

Winplot kjenner også igjen konstantene *pi* (3.14159265…) og *e* (2.71828…).

Å få mer hjelp

På hver meny i *Winplot*, og i mange dialogbokser, finnes det et valg som heter "Help…". Klikker du der, kommer det opp et vindu med informasjon om den aktuelle menyen eller dialogboksen.

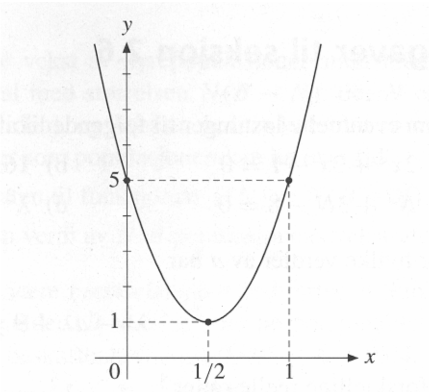
Kopiere et plott til *Word*

For å kopiere et plott til *Word* velger du først "File" -"Copy to clipboard" i plottevinduet. Ikke bruk "Bitmap to clipboard", det ser ikke ut til å fungere skikkelig. Hvis du vil ha med bakgrunnsfargen i plottevinduet, krysser du av på "File" - "With back color".

Deretter er det bare å sette markøren i *Word*-dokumentet der du vil ha plottet, og lime inn.

Endre koordinatsystemet i et plott

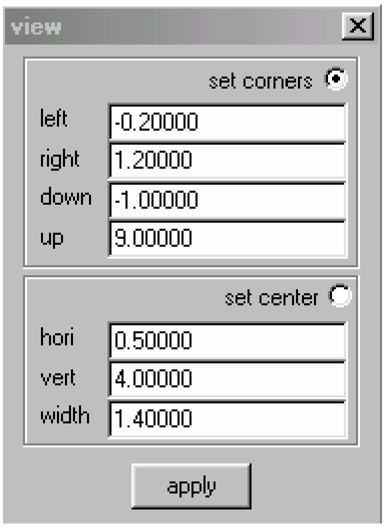
Det er vanskelig å lese noe særlig informasjon ut av kurven vi plottet tidligere. Den ligger "sammenklemt", og vi har ingen enheter å lese av på aksene. Samme funksjon er vist i figuren under, i et utklipp fra Tor Gulliksen: *Matematikk i praksis* 2. opplag 2000. Denne figuren gir mye mer informasjon fordi enhetene på koordinataksene er fordelt annerledes, og det er angitt verdier på aksene:



Figur 2.23

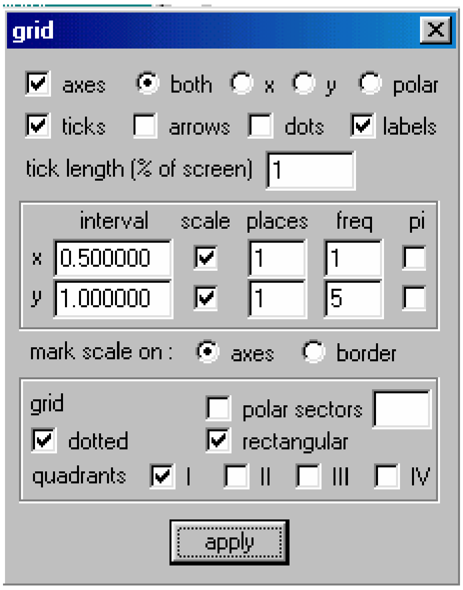
Vi skal nå prøve å endre på vårt plott, så det blir mer informativt.

I figuren over er det vist et utsnitt av *x*-aksen som går fra litt under 0, til litt over 1. Utsnittet av *y*-aksen går fra litt under 0 til ca. 9. Vi skal justere aksene i vårt plott tilsvarende. Velg "View" - "View…" i plottevinduet. I dialogboksen som kommer opp, angir du at vi vil se *x*-aksen i intervallet [-0.2, 1.2], og *y*-aksen i intervallet [-1.0, 9.0], som vist under:



Merk av "set corners", og klikk på "apply".

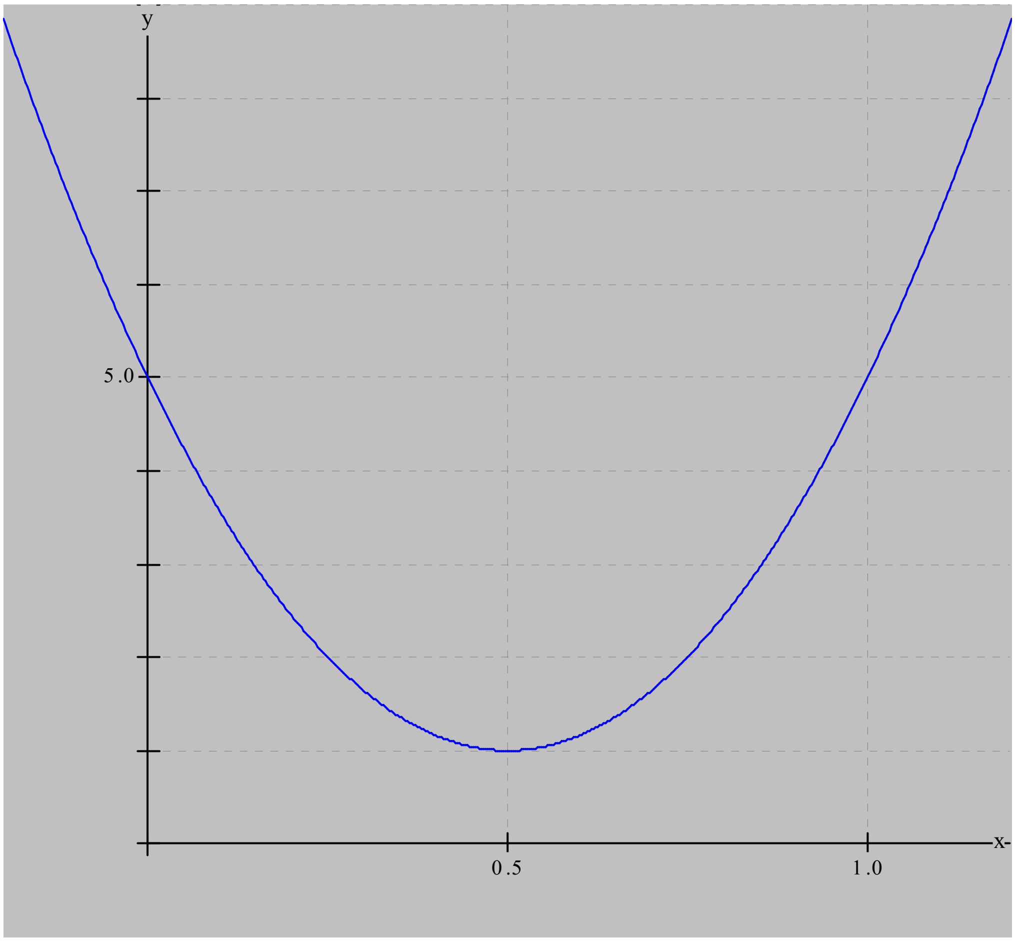
Så gjelder det å justere enhetene på aksene. Vi ønsker enhetsmerker for hver halve enhet på *x*-aksen, og hver enhet på *y*-aksen. Men på *y*-aksen ønsker vi bare hver femte enhet skrevet ut med tall. Vi ønsker videre å navngi *x*- og *y*-aksen, og å få tegnet inn et rutenett i første kvadrant. Det gjør vi ved å velge "View" - "Grid" i plottevinduet, og fylle ut som vist under:



Klikk "apply" når du er ferdig med å fylle ut.

Det er i et senere avsnitt detaljert beskrevet hva de forskjellige valgene i dette vinduet betyr. Eksperimenter, og se hvordan det påvirker plottet.

Under er plottet vist, slik det skal se ut hvis du har fulgt instruksjonene riktig:

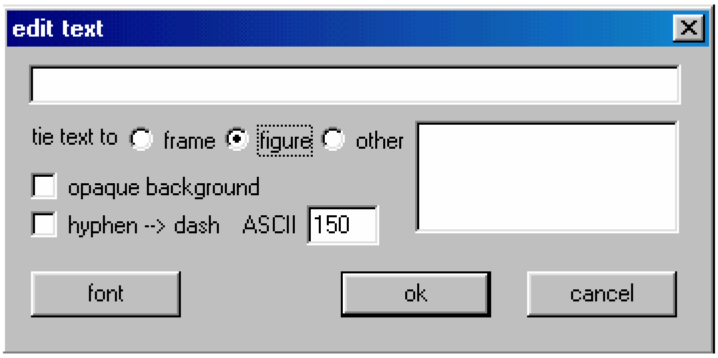


Layoutmessige justeringer, som å endre tykkelse og farge på aksene eller rutenett, bakgrunnsfarge, etc., gjøres fra "Misc"-menyen i plottevinduet. Den vil ikke bli nærmere beskrevet.

Prøv deg fram. Hvordan en endrer fargen på selve kurven, er beskrevet i avsnittet "Flere kurver i samme plott".

Skrive inn tekst

Det er enkelt å skrive inn tekst i et plott. Kryss av på "Btns" - "Text" i plottevinduet. Når du så høyreklikker i plottevinduet, kommer boksen under opp:



Skriv inn det du vil, og klikk "ok". Vil du senere editere, kan du høyreklikke på teksten i plottevinduet.

Vær oppmerksom på at hvis ikke " Btns" - "Text" er krysset av, vil høyreklikking bare føre til at plottet flytter seg i vinduet.

En del av valgene på "view"-menyen

Dette avsnittet gir en kjapp oversikt over de viktigste valgene på "view"-menyen i plottevinduet. De er nevnt i den rekkefølgen du bør lære dem å kjenne, ikke i den rekkefølgen de opptrer på selve menyen.

Det kan være nyttig å vite at hvis du høyreklikker på et punkt i plottet, blir det punktet satt i sentrum av plottevinduet, så fremt det ikke er krysset av for " Btns" - "Text".

Last window

Ved å velge dette, kan du trinnvis gå tilbake til tidligere utseender, angre med andre ord. Meget nyttig hvis noe blir helt galt.

Factor

Angir hvor stor forstørrelse / forminsking en ønsker når en zoomer.

Zoom in / Zoom out

Forstørrer / forminsker bildet så mye som angitt i "Factor".

View…

Brukes til å bestemme hvilken del av koordinatsystemet som skal vises i plottevinduet. Du kan enten velge å angi koordinatene til de fire hjørnene, "set corners", slik vi gjorde i et eksempel tidligere. Alternativet er å velge hvilket punkt i koordinatsystemet som skal ligge midt i vinduet, med "set center".

Zoom square

Hvis skalaen på de to aksene ikke er like (slik at sirkler ikke ser runde ut), kan du bruke dette valget for å gjøre rutenettet kvadratisk.

Grid…

Brukes til å sette enhetsmerker, verdier, navn, etc. på aksene, slik som vist i eksemplet tidligere. De enkelte valgene er:

- axes: Om akser skal vises eller ikke.
- both / x / y / polar. Om en skal vise begge aksene, bare *x*-aksen, bare *y*-aksen, eller om et polart koordinatsystem skal vises. (Plotting i polarkoordinater blir omtalt i avsnittet "*Winplot* - polarkoordinater").
- ticks: Setter små streker på aksene for hvert intervall.
- arrows: Setter pil på enden av hver akse.
- dots: Merker av et bakgrunnsmønster basert på enhetene på aksene.
- labels: Setter navn på *x*- og *y*-aksen.
- tick length: Hvor brede intervallmerkene skal være.
- interval: Hvor tett intervallmerkene skal settes på *x*/*y*-aksen.
- scale: Om verdier skal skrives på *x*/*y*-aksen.
- places: Antall desimaler verdiene på *x*/*y*-aksen skal angis med.
- freq: Hvor tett verdiene skal settes på *x*/*y*-aksen. 1 betyr "hvert intervallmerke", 2 betyr "annet hvert intervallmerke", etc.
- pi: Om intervallene på *x*/*y*-aksen skal angis som multipler av π. Dette er nyttig hvis en plotter trigonometriske funksjoner.
- mark scale on axes/borders: Om verdiene skal skrives på aksene eller i kanten av plottet.
- grid polar sectors/rectangular: Om et rutenett skal tegnes i bakgrunnen av plottet, henholdsvis i polare sektorer og/eller et rektangulært nett. (Plotting i polarkoordinater blir omtalt i "Winplot 2").
- grid dottet: Om rutenettet i bakgrunnen skal vises som stiplede linjer.
- grid quadrants I/II/III/IV: Hvilke kvadranter rutenettet skal tegnes i. (Gjelder kun rektangulært nett)

Winplot - kurver

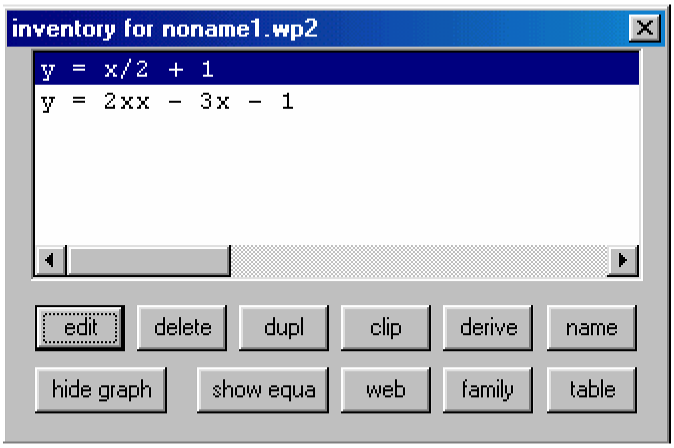
Dette avsnittet tar for seg tegning av flere kurver i samme plott, hvordan en leser av verdier på en kurve, finner skjæringspunktet mellom en kurve og aksene, eller mellom to kurver. Det vises hvordan en tegner en kurvefamilie, og hvordan en kan endre parametere ved hjelp av blafelt.

Flere kurver i samme plott

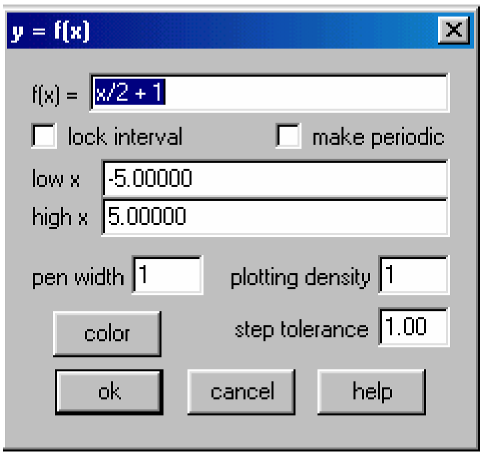
I avsnittet "*Winplot* - Introduksjon" så vi hvordan vi kunne plotte en kurve, og tilpasse koordinatsystemet. Vi skal nå ta dette et skritt videre, og tegne to kurver i samme koordinatsystem.

Start *Winplot*, og åpne et plottevindu, som beskrevet i avsnittet "Lage et plott". Så velger du "Equa" - "Explicit…" og skriver inn "x/2 + 1". Deretter velger du "Equa" - "Explicit…" på nytt, og skriver "2xx -3x -1". Du har nå laget et plott av funksjonene $y = \frac{x}{2} + 1$ og $y = 2x^2 - 3x - 1$.

Til et hvert plott tilordner *Winplot* et inventarvindu som lister opp funksjonene i plottet. Vårt inventarvindu ser nå slik ut:



I inventarvinduet kan en gjøre en gjøre forskjellige manipulasjoner på funksjonene. En markerer den funksjonen en vil arbeide med, og klikker på en av knappene, for eksempel "edit" for å endre på funksjonen, eller "delete" for å slette funksjonen fra plottet. Når en klikker "edit", får en opp et vindu likt det som er vist under, der en blant annet kan skifte farge på kurven:

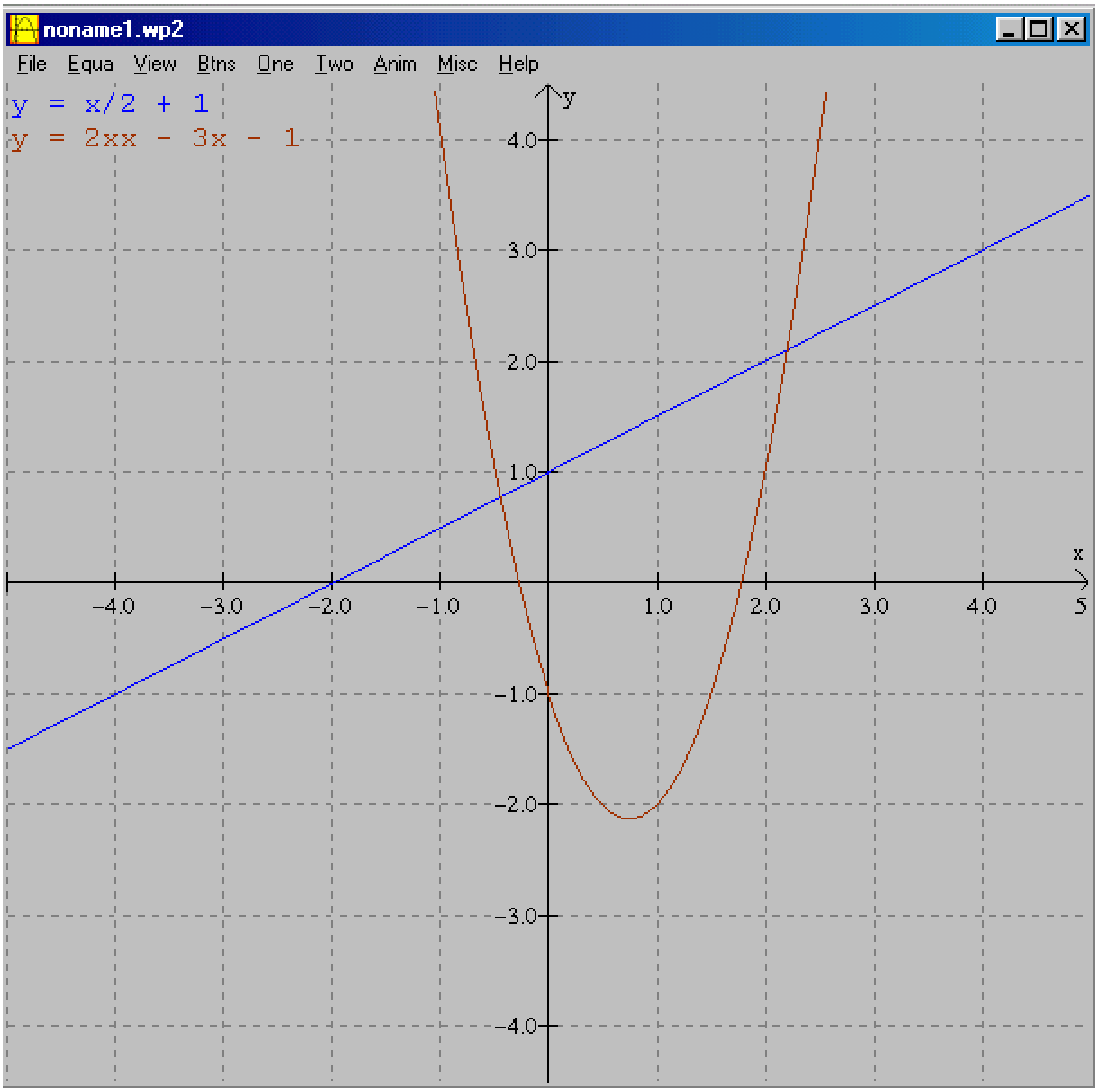


Sett fargen på $y = \frac{x}{2} + 1$ til blå, og $y = 2x^2 - 3x - 1$ til brun.

Skru på visning av funksjonsnavn ved å bruke knappen "show equa". Som du ser, vises funksjonsnavnene med samme farge som kurvene, slik at det er enkelt å se hva som er hva.

Dersom inventarvinduet skulle bli borte, kan du hente det fram igjen med "Equa" - "Inventory…" på menyen i plottevinduet.

Plottet skal se omlag slik ut:



Klikk på "edit" igjen i inventarvinduet. Der ser du at noe som heter "low x" er satt til -5, og "high x" til 5. Dette angir definisjonsområdet til en funksjon i *Winplot*, i vårt tilfelle $-5.0 \leq x \leq 5.0$. Definisjonsområdet kan endres ved å krysse av på "lock interval" og skrive inn nye verdier for "low x" og "high x".

Med definisjonsområdet mener vi her det *x*-området vi ber *Winplot* forholde seg til. I virkeligheten er jo begge funksjonene definert for alle verdier av *x*.

Prøv å endre definisjonsområdet for $y = \frac{x}{2} + 1$ til [-2.0, 2.0], så til [-10.0, 10.0], og deretter tilbake til [-5.0, 5.0].

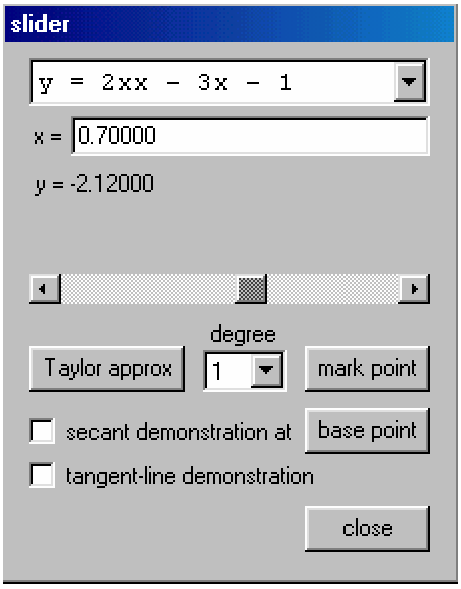
Som du sikkert så, ble grafen klippet av da definisjonsområdet ble satt til [-2.0, 2.0], mens det ikke var noen synlig forskjell på [-5.0, 5.0] og [-10.0, 10.0]. Det skyldes at denne utvidelsen i definisjonsområdet havner utenfor plottevinduet. Vil du se mer av grafen, må du også endre i "view"-vinduet, som beskrevet i avsnittet "Endre koordinatsystemet i et plott".

Lese av verdier på en kurve

I *Winplot* kan en la en markør gli langs en kurve, og lese av de tilhørende *y*-verdiene.

I plottevinduet finnes et menyvalg som heter "One". Under denne menyen finnes forskjellige ting en kan gjøre med en enkelt kurve. Under "Two" finnes ting en kan gjøre med to kurver samtidig.

Velg "One" - "Slider". Øverst i vinduet som kommer opp kan du velge hvilken funksjon du vil se på. Velg $y = 2x^2 - 3x - 1$ som vist under:



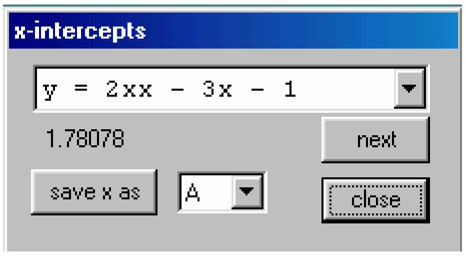
Skyv på blafeltet, og du vil se et trådkors følge kurven i plottevinduet. Trådkorsets posisjon vises i slider-vinduet. I slider-vinduet vist over ser vi at trådkorset står i punktet $(0.70000, -0.212000)$.

Det går også an å skrive inn en *x*-verdi direkte i slider-vinduet. Prøv "0" og "sqrt(2)". Trykk <enter> for å få *Winplot* til å svelge det du har skrevet.

Klikker en på "mark point", settes det et punkt der trådkorset står. Dette punktet kommer opp som et eget innslag i inventarvinduet, og kan editeres og slettes derfra. Prøv å editere et punkt, og kryss av for "anchors". Det opprettes normaler inn på aksene, slik at punktets koordinater kan leses av i selve plottet.

Finne nullpunkter til en kurve

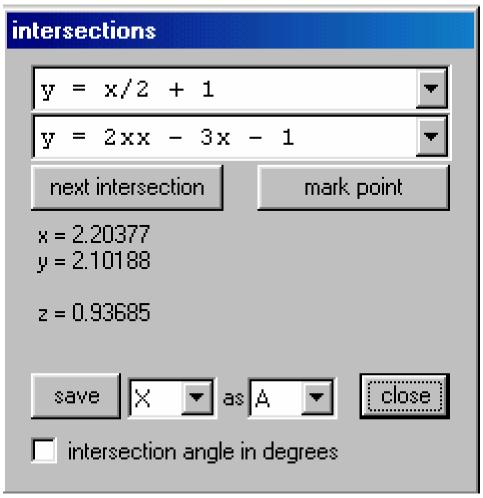
For å finne ut hvor en kurve skjærer x-aksen, velger du "One" - "Zeros..." i plottevinduet. Et vindu tilsvarende slider-vinduet kommer opp, som vist under:



Dette forteller at kurven skjærer x-aksen omlag i punktet (1.78078, 0). Trykker du "next", finner du det andre skjæringspunktet, (-0.28078 ,0). Dette betyr at løsningene på annengrads-ligningen $2x^2 - 3x - 1 = 0$, er $x_1 \approx 1.78078$ og $x_2 \approx -0.28078$.

Finne skjæringspunkter mellom to kurver

For å finne ut hvor to kurver skjærer hverandre, velger du "Two" - "Intersections...", og får opp et vindu likt det under:



Siden vi bare har to kurver i vårt plott, er de allerede valgt for oss. Har en flere kurver, kan det være at en må bruke pilene for å velge riktige kurve. Vi ser at $y = \frac{x}{2} + 1$ og $y = 2x^2 - 3x - 1$ skjærer hverandre i omlag (2.20377, 2.10188). Klikker du "next" får du fram et annet skjæ-ringspunkt, (-0.45377, 0.77312). Med andre ord er løsningene på ligningen $2x^2 - 3x - 1 = \frac{x}{2} + 1$: $x_1 \approx 2.20377$ og $x_2 \approx -0.45377$, og de tilhørende *y*-verdiene $y_1 \approx 2.10188$ og $y_2 \approx 0.77312$. (z-verdien som vises har ingen mening, det må være en feil i *Winplot*).

Også i dette vinduet kan en velge "mark point", og få opp skjæringspunktene som innslag i inventarvinduet.

Winplot angir bare omtrentlige verdier. Ved å regne ut med penn og papir, kan vi finne de nøyaktige skjæringspunktene: $\left(\frac{7+\sqrt{113}}{8}, \frac{23+\sqrt{113}}{16}\right)$ og $\left(\frac{7-\sqrt{113}}{8}, \frac{23-\sqrt{113}}{16}\right)$.

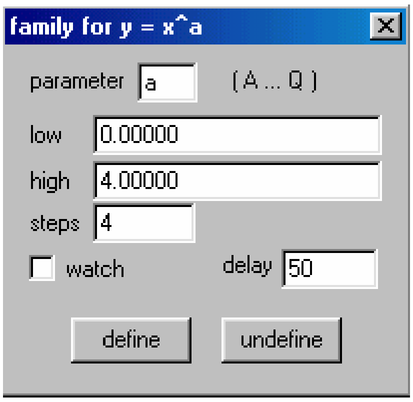
Plotte en kurvefamilie

Vi starter på ny frisk, og åpner et nytt plottevindu.

Vi ønsker å plotte funksjonene $y = x^0$, $y = x^1$, $y = x^2$, $y = x^3$ og $y = x^4$ i samme diagram.

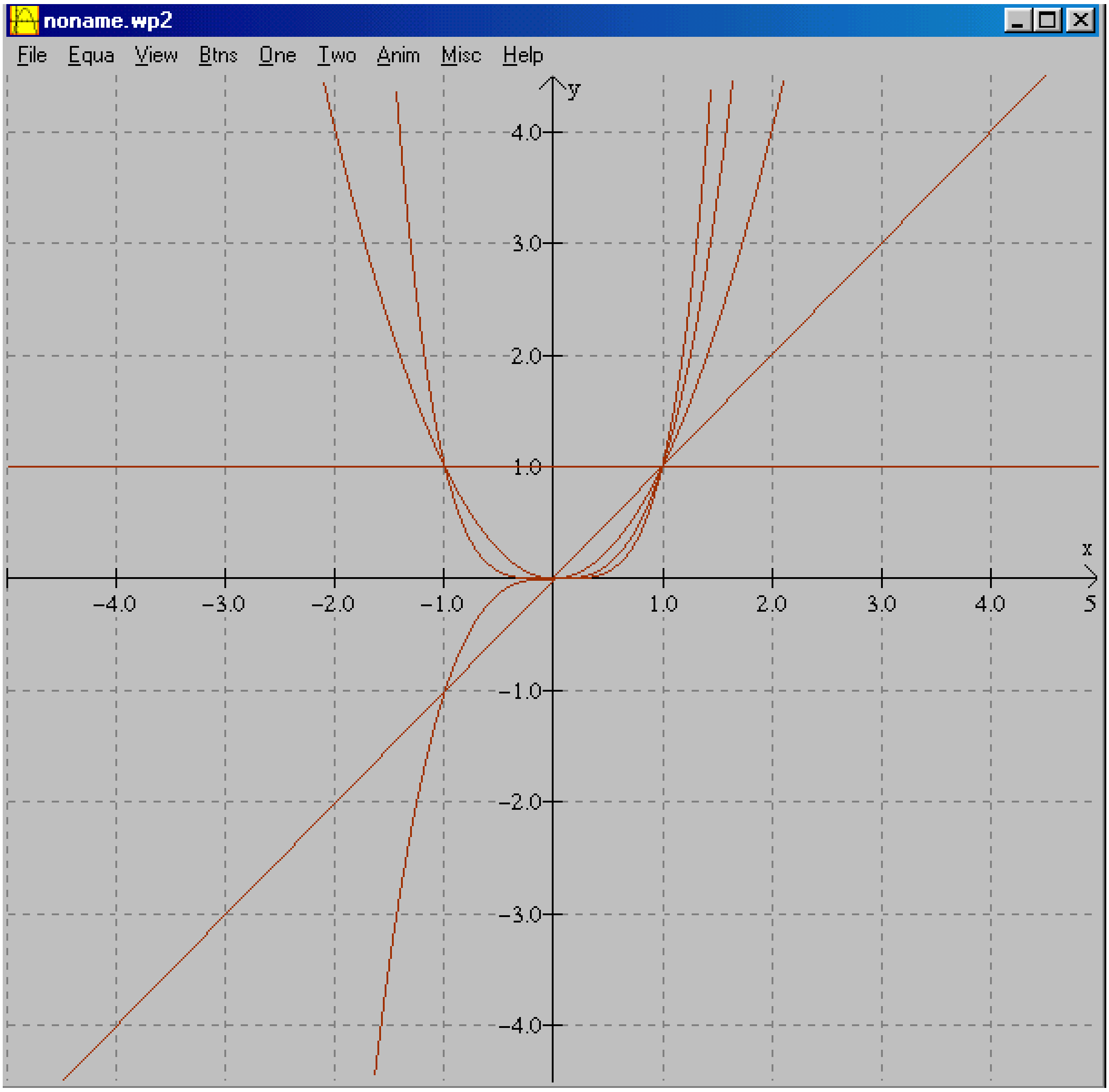
Det er fullt mulig å skrive inn disse funksjonene én for én, men i dette tilfellet kan vi gjøre det på en enklere måte, ved å plotte en kurvefamilie. Alle de nevnte funksjonene kan nemlig skrives på formen $y = x^a$, der $a \in [0,1,2,3,4]$.

Velg "Equa" - "explicit..." i plottevinduet som tidligere, skriv inn "x^a", og klikk "ok". Klikk så på "family" i inventarvinduet. Familievinduet kommer opp. Skriv inn at parameteren *a* skal gå fra 0.0 til 4.0 i fire skritt, som vist under:



Winplot tolker dette slik at den skal ta fire like lange skritt med utgangspunkt i verdien 0.0. Lengden på skrittene må være slik at siste skritt ender på 4.0. Vi får med andre ord de fem verdiene 0.0, 1.0, 2.0, 3.0 og 4.0. Dette er forferdelig klønete.

Klikk på "define", og *Winplot* beregner *a*-verdiene, setter dem inn én for én inn i ligningen $y = x^a$ og plotter de tilhørende diagrammene, som vist under:



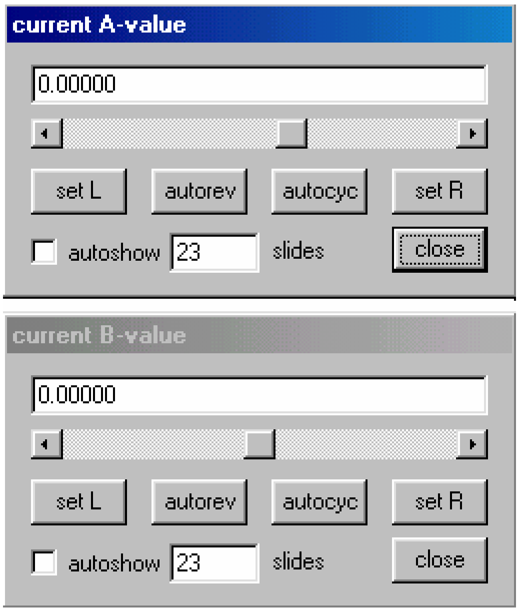
Winplot bruker dessverre ikke forskjellig farge på kurvene, så det kan være vanskelig å se hva som er hva.

Endre parametere med blafelt

I *Winplot* er det mulig å legge inn parametere som en kan justere ved hjelp av blafelt. Vi skal illustrere dette ved å plotte $y = f(x-a)+b$, for funksjonen $f(x) = x^2$. Vi ønsker å finne ut hva som skjer med kurven når *a* og *b* endres.

Åpne et nytt plottevindu, velg "Equa" - "explicit... som før, og skriv inn "(x-a)^2+b".

Gå nå på "Anim"-menyen i plottevinduet og velg først "A…", så "B…". Du får opp to vinduer som vist under:



Du kan nå justere på blafeltene, og se hvordan kurven flytter seg.

I utgangspunktet varierer parametrene mellom -10 og 10, men det kan du selv endre. Skriv inn et tall og trykk "set L" for nedre grenseverdi, og "set R" for øvre grenseverdi.

Hvis du klikker på "autorev" eller "autocyc" vil parameteren automatisk løpe fram og tilbake mellom grenseverdiene. OBS! For å skru av "autorev" og "autocyc" trykker du <ctrl>q.

Som du kan se, har "Misc"-menyen valg fra "A" til "W". Det går altså an å bruke opp til 23 parametere samtidig.

Winplot - mer om kurver

Dette avsnittet tar for seg kopiering av kurver, speiling av kurver, og plotting av implisitte funksjoner.

Kopiering og modifisering av kurver

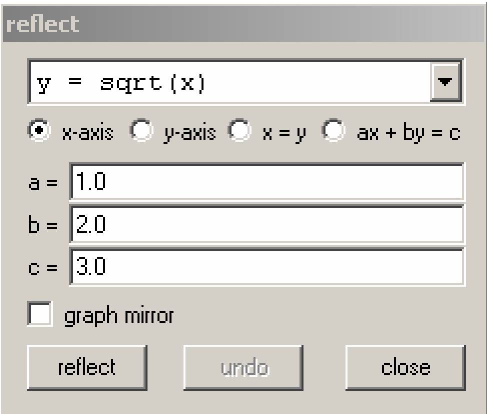
Av og til kan det være at en skal plotte kurver basert på ligninger som er nesten like, for eksempel $y_1 = \dfrac{16-2x^2}{\sqrt{16-x^2}}$ og $y_2 = \dfrac{16-2x^2}{\sqrt{16+x^2}}$. I *Winplot* blir uttrykket for

y_1: (16 - 2 x^2) / sqrt(16 - x^2), og for y_2: (16 - 2 x^2) / sqrt(16 + x^2).

Alt er likt i de to likningene, bortsett fra at et minustegn er endret til et plusstegn. Heldigvis er det ikke nødvendig å skrive inn hele formelen på nytt. Når formelen for y_1 først er skrevet, kan en gå i inventarvinduet og klikke på "dupl". På spørsmål om en vil slette den opprinnelige kurven, svarer en "no", og deretter kommer det opp en et vindu der en kan editere på en kopi av ligningen. I eksemplet over endrer en minustegnet til et plusstegn, trykker "ok", og kurven til y_2 vises sammen med kurven til y_1.

Speiling av kurver

Winplot gir mulighet for å speile kurver. Velger en "One", "Reflect…" i plottevinduet, får en opp et vindu som vist under:



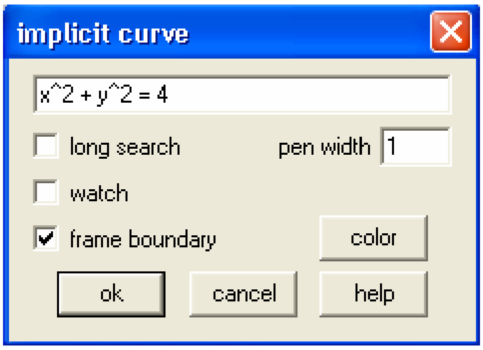
En må på forhånd ha plottet minst én kurve, i dette eksemplet kurven til funksjonen $y = \sqrt{x}$. Har en plottet flere kurver, velger en hvilken kurve en vil speile ved å trykke på knappen med pil-symbolet. Klikker en av for "graph mirror", skisserer *Winplot* linjen en speiler om.

En kan speile om *x*-aksen, *y*-aksen, linjen $x = y$, eller en vilkårlig linje $ax + by = c$, der en selv bestemmer parametrene *a*, *b* og *c*.

En speiling om linjen $x = y$ gir et plott av den inverse funksjonen, i dette eksemplet $y = x^2$.

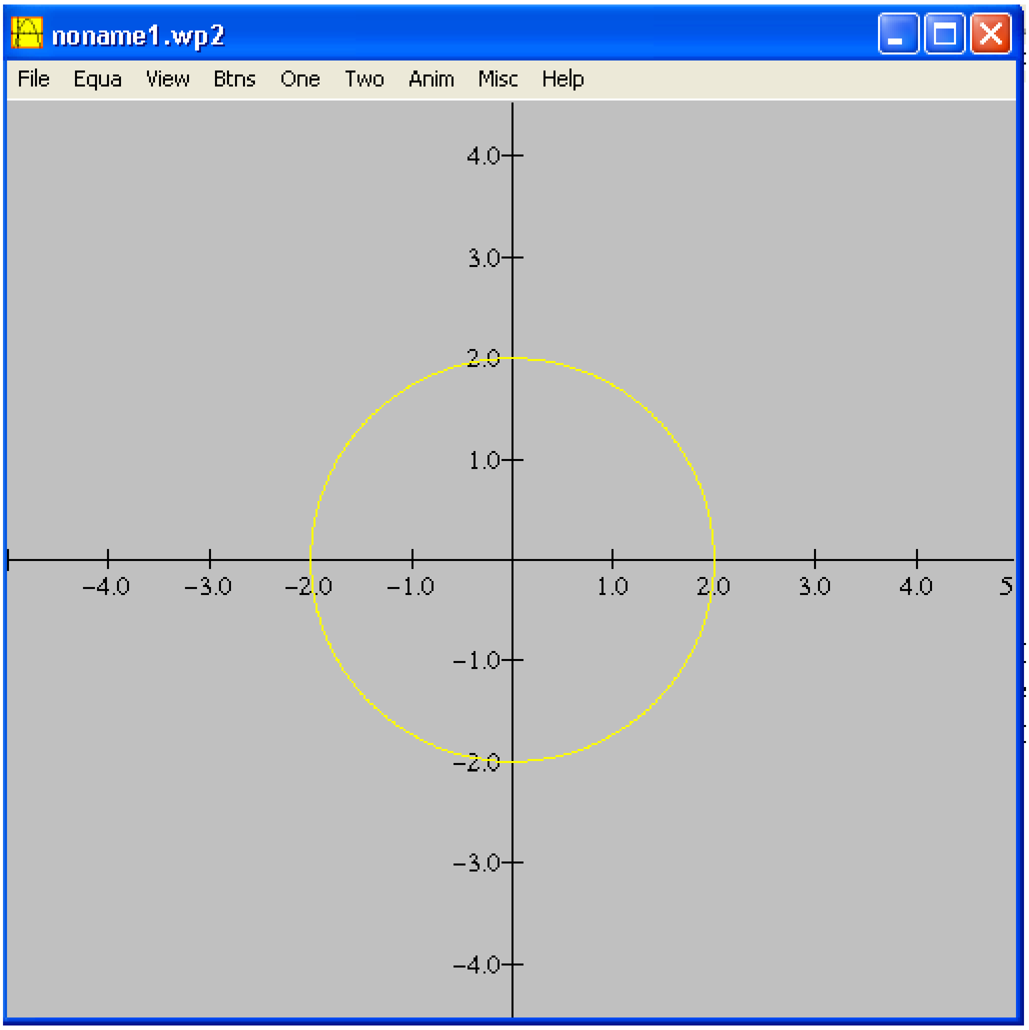
Plott av implisitte funksjoner

I de fleste beskrivelsene har vi plottet *y* som en eksplisitt funksjon av *x*, men i *Winplot* kan en også oppgi en funksjon implisitt. Det kreves da ikke at en har en enkelt *y* pent plassert på venstre side av likhetstegnet, og et uttrykk som bare avhenger av *x* på høyre side. En skriver bare inn funksjonen som den er. Som et eksempel skal vi plotte funksjonen $x^2 + y^2 = 4$. Vi velger "Equa", - "Implicit…", og i vinduet som kommer opp, fyller vi inn x^2 + y^2 = 4, som vist under:



Pass på at det er klikket av for "frame boundary" og ikke for "long search". I motsatt fall kan *Winplot* bli stående og arbeide lenge. Trykk i så fall <Ctrl>q for å få den til å stoppe. De som er interessert i hva disse valgene betyr, eller hvordan *Winplot* arbeider for å plotte en implisitt funksjon, kan velge "Equa" - "Help…", og bla seg ned til "Implicit".

Når vi klikker "ok", skisserer *Winplot* kurven, som ser omlag slik ut:



En sirkel med sentrum i origo og radius 2.

En funksjon som er oppgitt implisitt, kan editeres, kopieres og speiles i *Winplot*, på samme måte som eksplisitte funksjoner. Men *Winplot* er dessverre ikke i stand til å finne en implisitt funksjons nullpunkter eller ekstremalpunkter. Ei heller kan den derivere, eller tegne tangenter til en implisitt funksjon.

I eksemplet over kunne vi lett ha omarbeidet den implisitte ligningen til to eksplisitte:

$y_1 = \sqrt{4 - x^2}$ og $y_2 = -\sqrt{4 - x^2}$, to halvsirkler. Så kunne vi plottet dem hver for seg på vanlig måte. Men ofte støter en på funksjoner som er vanskelige eller umulige å uttrykke eksplisitt, og da kan muligheten for å plotte implisitte funksjoner i *Winplot* gi verdifull informasjon om hvordan kurven ser ut.

Winplot - trigonometriske funksjoner

Dette avsnittet tar for seg et par ting som er greit å kunne når en plotter trigonometriske funksjoner, og det vises et eksempel på at *Winplot* langt fra er ufeilbarlig.

Plotting av trigonometriske funksjoner

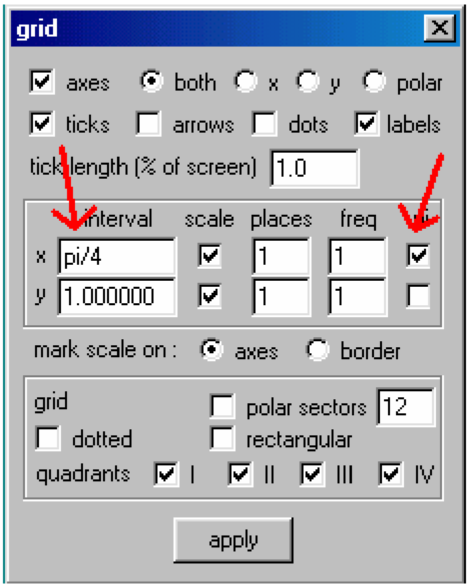
Å plotte trigonometriske funksjoner gjør en på samme måte som ved andre funksjoner. La oss si at vi ønsker å lage et plott av $\sin(x)$, $\cos(x)$ og $\tan(x)$ i samme diagram.

Velg "Equa" - "Explicit…" og skriv inn "sin(x)". Velg så "Equa" - "Explicit…" på nytt, og skriv "cos(x)". Tilsvarende gjør du for tan(x).

Gå i inventarvinduet, skru på visning av funksjonsnavnene ved å velge funksjonene etter tur, og klikke på "show equa". Endre samtidig fargen på kurvene ved å velge "edit" - "color". Gjør kurven til $\sin(x)$ blå, $\cos(x)$ rød, og $\tan(x)$ mørkegrønn.

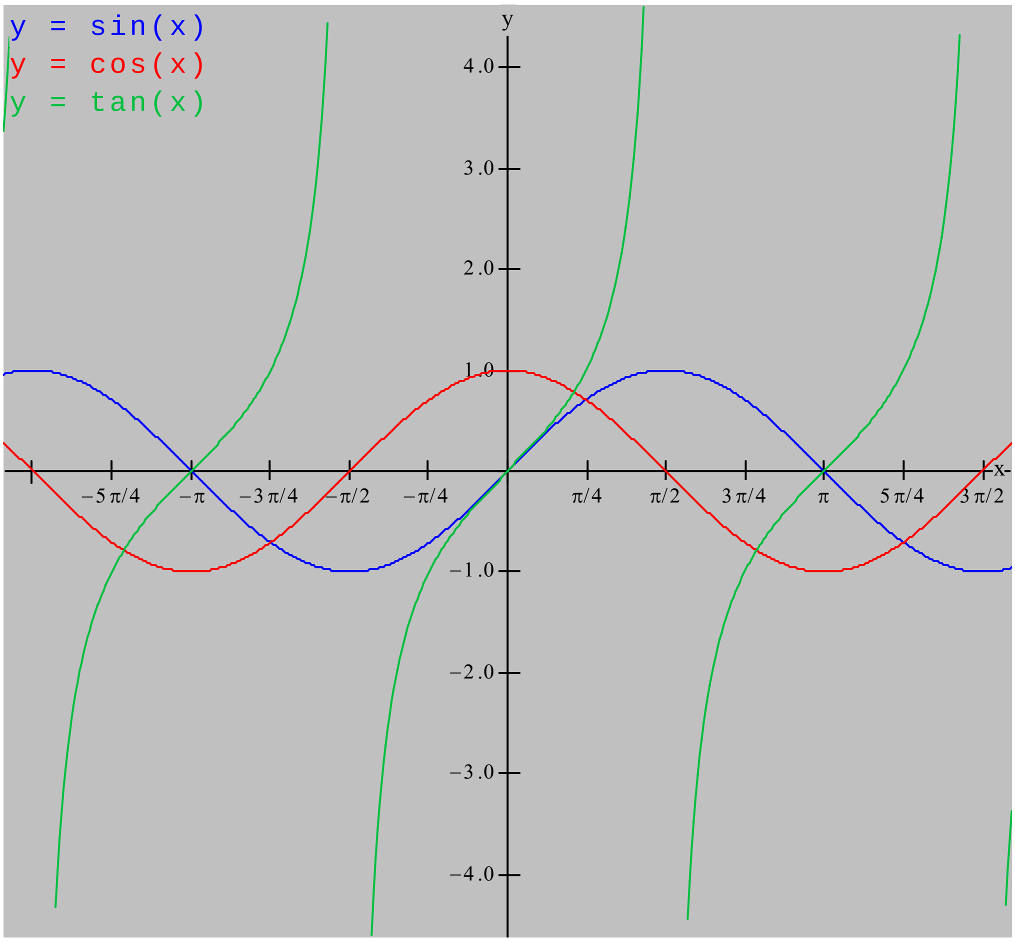
Så kommer det som er lurt å vite når en skal plotte trigonometriske funksjonene. Nesten alle dataprogrammer som har trigonometriske funksjoner innebygd, forventer at x oppgis i radianer, ikke i grader. Siden 90° tilsvarer $\frac{\pi}{2}$, 180° tilsvarer π etc., vil det i et plott av trigonometriske funksjoner derfor være naturlig å velge et multiplum av π som enhet på x-aksen.

Hent fram grid-dialogboksen, velg $\frac{\pi}{4}$ som enhet på x-aksen og kryss av for "pi", som vist under:



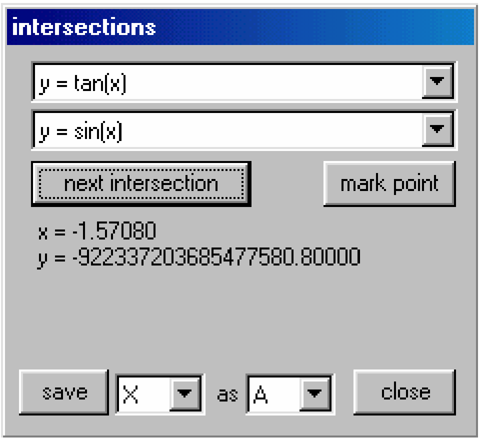
Legg merke til at det faktisk går an å skrive "pi" med bokstaver.

Det resulterende plottet ser omlag slik ut:



Winplot er ikke ufeilbarlig

Prøv å bruke *Winplot* til å finne skjæringspunktene mellom kurvene til $\sin(x)$ og $\tan(x)$. Velg "Two" - "Intersections" i plottevinduet. La første funksjon være $\tan(x)$ og andre funksjon $\sin(x)$ som vist under. Klikk på "next intersection" for å se de forskjellige skjæringspunktene.



Winplot forteller ganske riktig at kurvene skjærer hverandre i $(-\pi,0)$, $(0,0)$ og $(\pi,0)$, men kommer også med noen underlige forslag, som (-1.57080, -922337203685477580.8000).

Det er et helt absurd resultat, verdien til $y = \sin(x)$ beveger seg jo aldri utenfor området [-1.0, 1.0].

Ser vi litt nærmere på hvor det går galt, ser vi at *x*-verdien i disse tilfellene er nokså nærme en verdi der $\tan(x)$ ikke er definert, men har en vertikal asymptote. For eksempel

$$-1.57080 \approx -\frac{\pi}{2}.$$

y-verdien akselererer mot $+\infty$ på den ene siden, og $-\infty$ på den andre siden. "Uendelig store" tall, og verdier som endrer seg svært raskt, er vanskelig å representere skikkelig i et dataprogram. Det samme er verdier som kommer veldig nærme null.

Vær derfor oppmerksom når slike fenomener opptrer, og svelg ikke alle resultater rått.

Men vi tilgir *Winplot*, det er tross alt gratis.

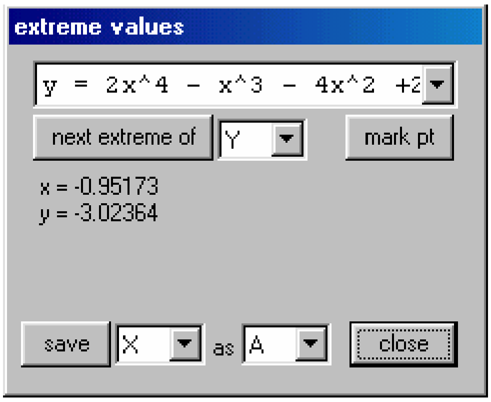
Winplot - Derivasjon

Dette avsnittet tar for seg hvordan en finner ekstremalpunktene til en funksjon, tegner tangenter til en kurve, og tegner kurven til den deriverte av en funksjon.

Finne ekstremalpunkter til en kurve

Vi starter med å plotte kurven $y = 2x^4 - x^3 - 4x^2 + 2x$. Velg "Equa" - "Explicit…" og skriv inn "2x^4 - x^3 - 4x^2 +2x".

Velger så "One" - "Extremes..." i plottevinduet. Et vindu som vist under, kommer opp:



Samtidig markeres ekstremalpunktet med et kryss i plottevinduet. Ved å klikke på "next extreme of", får en *Winplot* til å flytte til neste ekstremalpunkt. I vårt tilfelle er ekstremalpunktene omlag (-0.95173, -3.02364), (0,24221,0.24243) og (1.08452, -1.04447). *Winplot* forteller ikke noe om det dreier seg om globale eller lokale ekstremalpunkter, eller om det er minimums- eller maksimumspunkter.

Når en klikker på "mark point", blir ekstremalpunktet markert, og det dukker opp som innslag i inventarvinduet, der det kan editeres.

Tegne tangent til en kurve

Velger du "One" - "Slider..." får du opp slider-vinduet, som beskrevet i veiledningen *Winplot kurver*.

Krysser du av for "tangent-line demonstration" i slider-vinduet, vil *Winplot* hele tiden vise tangenten til kurven der trådkorset står. Tangentens stigningstall (slope) vises hele tiden i slider-vinduet.

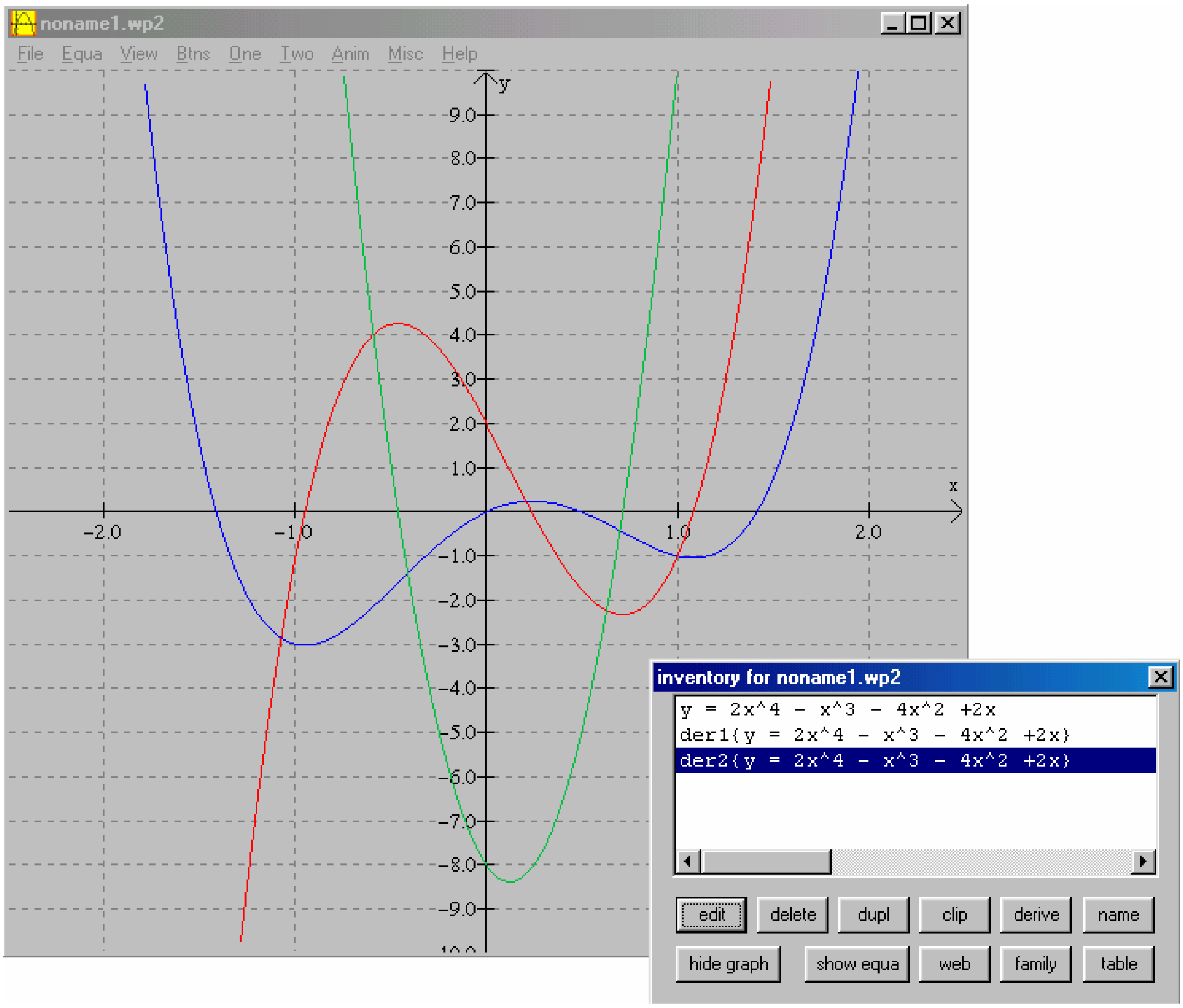
Derivasjon

I inventarvinduet finnes en knapp som heter "derive". Klikker du på den, tegnes kurven til den deriverte av en funksjon i plottevinduet. Den deriverte kommer opp som et eget innslag i inventarvinduet, der den kan editeres, for eksempel ved å skifte farge.

Den deriverte kan selv deriveres, og så videre, slik at en lett kan tegne den annenderiverte, tredjederiverte, etc. til en funksjon.

De deriverte kan undersøkes for nullpunkter, skjæringspunkter og ekstremalpunkter på linje med alle andre funksjoner.

Bildet under viser et plott av $y = 2x^4 - x^3 - 4x^2 + 2x$ i blått, y' i rødt, og y'' i mørkegrønt, og det tilhørende inventarvinduet.



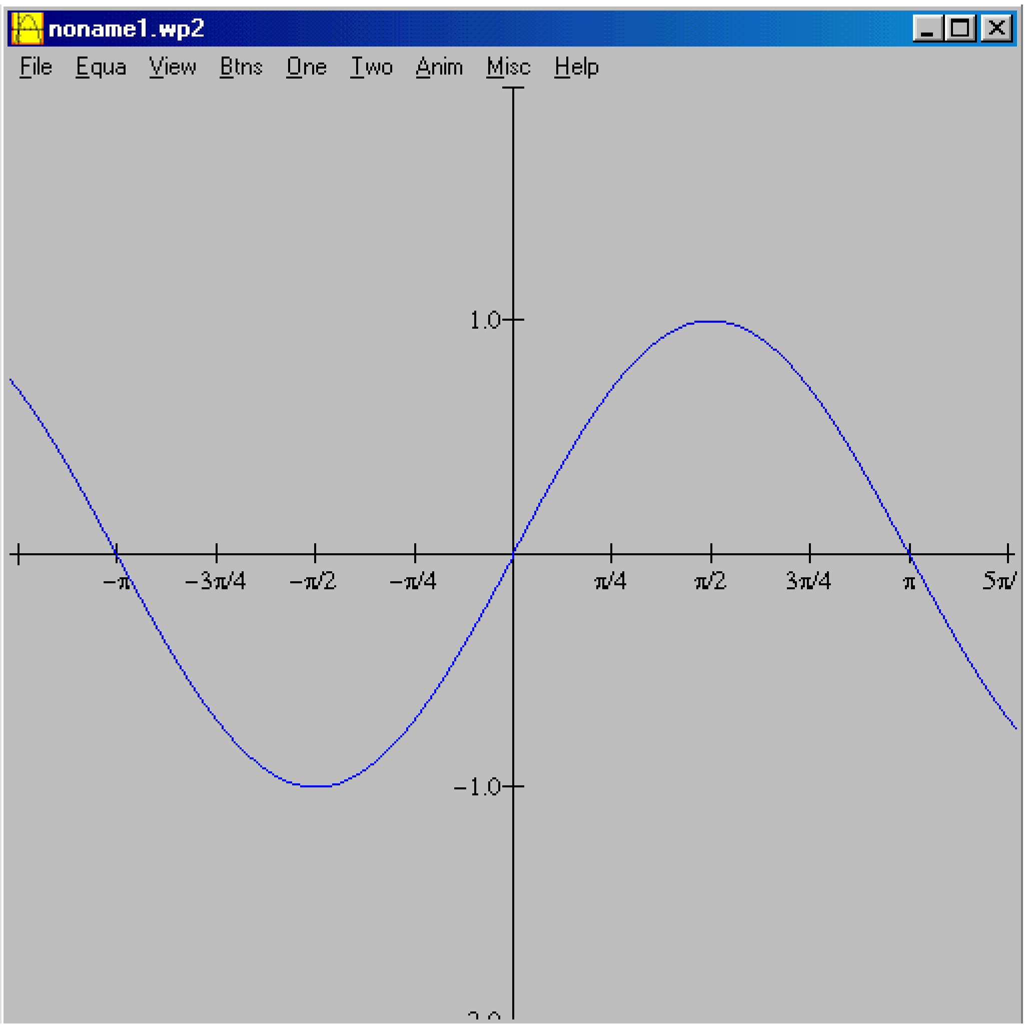
Legg merke til at den røde kurven, y', skjærer x-aksen når den blå kurven, y, har ekstremal-punkter. Den grønne kurven, y'', skjærer x-aksen når den blå kurven, y, har vendepunkter.

Winplot - integrasjon

Dette avsnittet tar for seg bruk av *Winplot* i forbindelse med bestemte og ubestemte integraler.

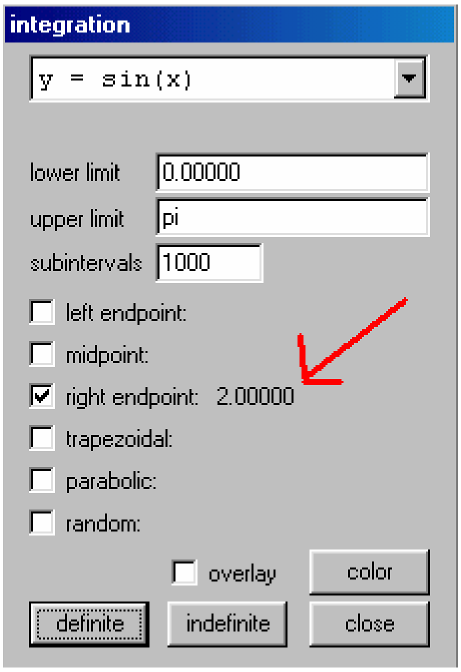
Bestemte integraler

Til arbeidet med bestemte integraler tar vi utgangspunkt i funksjonen $y = \sin(x)$, vist i plottet under:



Vi vil beregne det bestemte integralet: $\int_0^{\pi} \sin(x)dx$, ved hjelp av *Winplot*.

I plottevinduet velger vi "One" - "Mesasurement ►" - "Integration…" og får opp integrasjonsvinduet. Vi fyller ut "lower limit" og "upper limit" med integrasjonsgrensene, dvs. 0 og π, krysser av for "right endpoint", og klikker på "definite". *Winplot* utfører integrasjonen, og kommer fram til at svaret er 2.00000, som vist under:



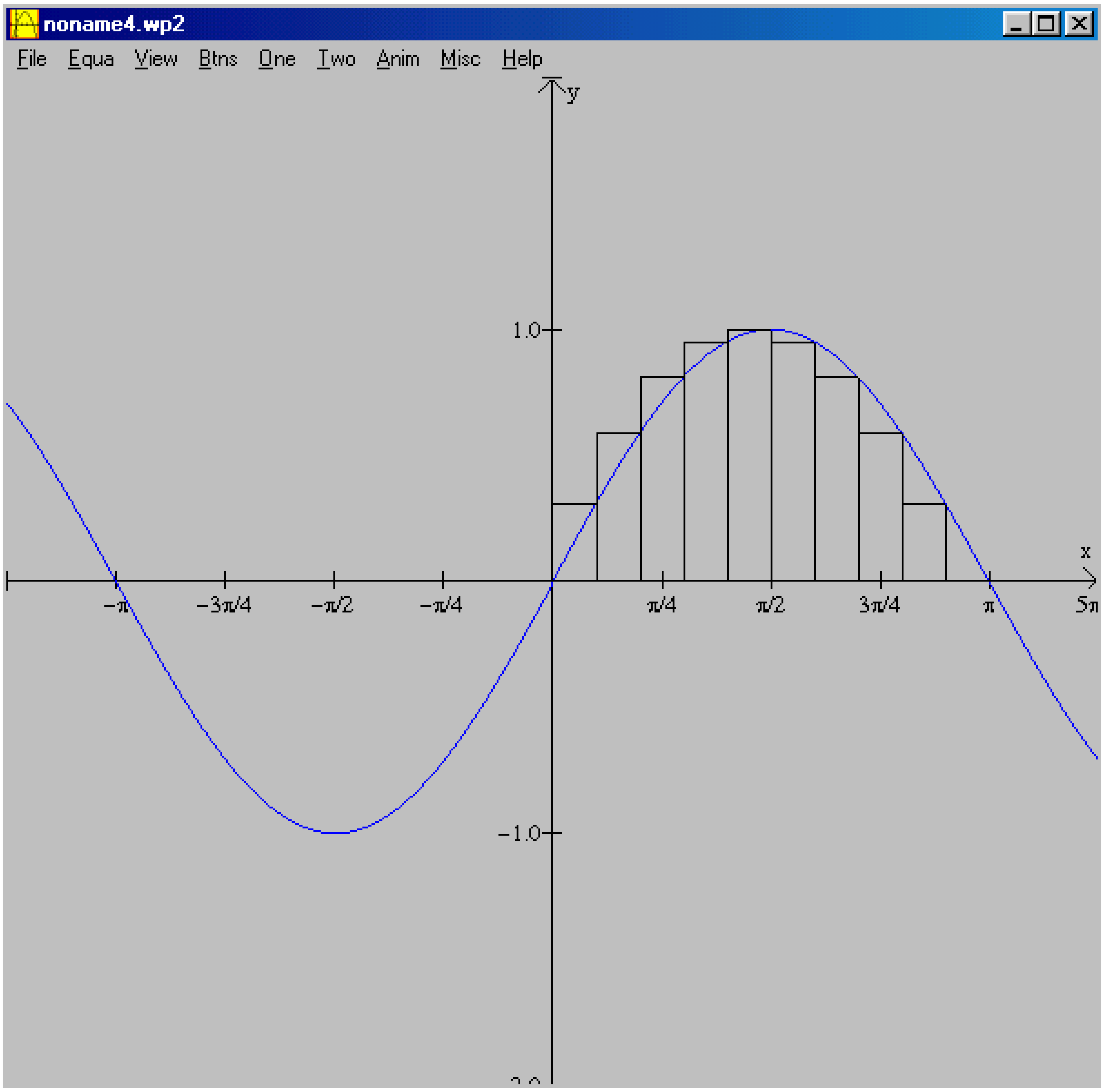
Vi kan lett verifisere at svaret er riktig: $\int_0^{\pi} \sin(x)dx = \Big|_0^{\pi} -\cos(x) = -(-1-1) = \underline{\underline{2}}$

I dette tilfellet var det enkelt å beregne integralet for hånd. Men mange integraler en støter på i praksis, er slett ikke lette å løse, og da kommer dataprogrammer som *Winplot* til sin rett. $\int_a^b e^{-x^2} dx$ er et eksempel på et slikt integral.

Bestemte integraler som areal

I lærebøker i matematikk vises det hvordan det bestemte integralet til en funksjon kan tolkes som arealet under funksjonens kurve, og det hvordan en tilnærmet kan beregne dette arealet ved å dele det opp i rektangler. Det er nettopp det *Winplot* gjør. Hent fram integrasjonsvinduet igjen, og sett "subintervals" til 10. "subintervals" angir hvor mange rektangler *Winplot* skal bruke. Trykk på "definite". Nå kommer *Winplot* fram til et litt galt svar, 1,98352. Feilen skyldes at oppdelingen av arealet i rektangler er for grov.

Kryss av for "overlay", og trykk på "definite" på nytt. *Winplot* tegner rektanglene inn i plottet, som nå ser omlag slik ut:

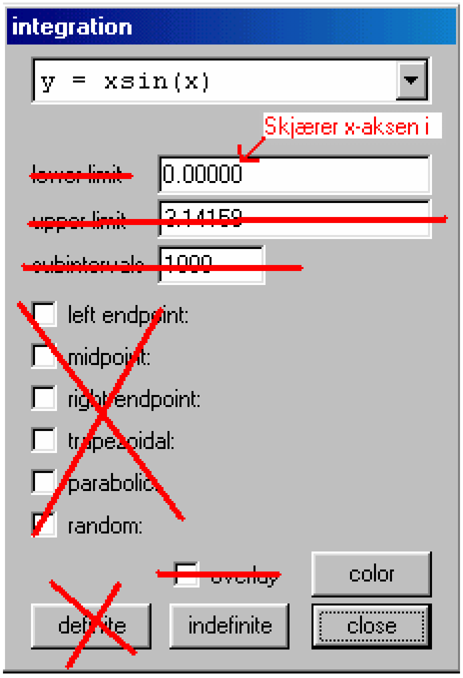


Vi har krysset av for "right endpoint", som betyr at rektanglene legges slik at høyre hjørne berører kurven. *Winplot* gir også mulighet for å bruke venstre hjørne, eller midtpunktet. ("left endpoint" og "midpoint"). Prøv disse mulighetene og iaktta plottet. Pass på å bare krysse av for en av gangen. Hvilken gir mest nøyaktig svar? Vil samme metode gir mest nøyaktig svar også for en kurve som ikke er symmetrisk som denne?

Det er også mulig å bruke andre former enn rektangler. "trapezodial", bruker en trapes-form, og "parabolic" en parabel mellom de to hjørnepunktene. "parabolic" representerer det som kalles *Simpsons regel*.

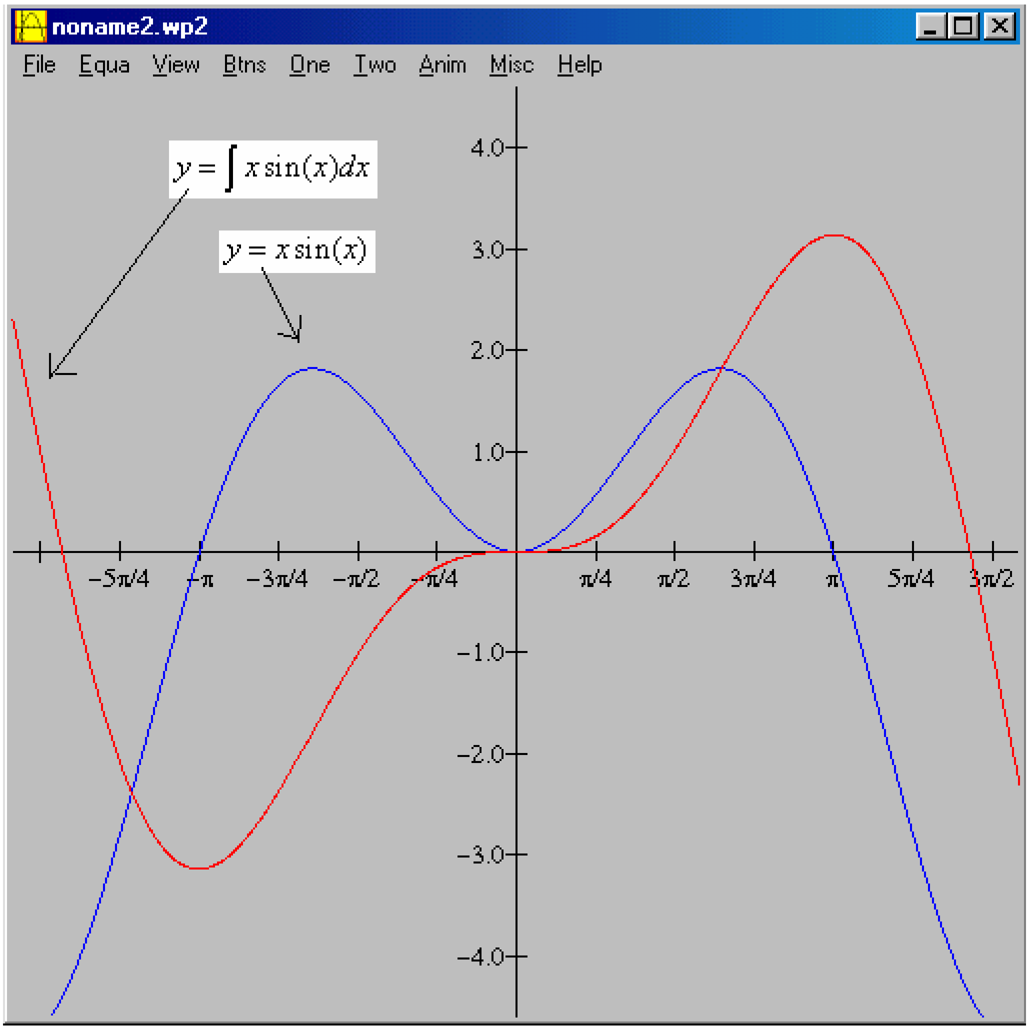
Ubestemte integraler

Winplot kan også skissere kurven til et ubestemt integral. Vi velger på samme måte som tidligere "One" - "Integration ►" - "Integration…". *Winplot* er litt rotete her, for samme vindu som ved bestemt integrasjon brukes, og de fleste valgmulighetene har ingen betydning i dette tilfellet. De er strøket over i bildet under:



La oss som eksempel plotte funksjonen $y = x\sin(x)$ og et tilhørende ubestemt integral.

Vi plotter $y = x\sin(x)$ på vanlig måte, og klikker på "indefinite" i integrasjonsvinduet. Kurven til det ubestemte integralet dukker opp i plottevinduet, og kommer også som et innslag i inventarvinduet. Det resulterende plottet er vist under:

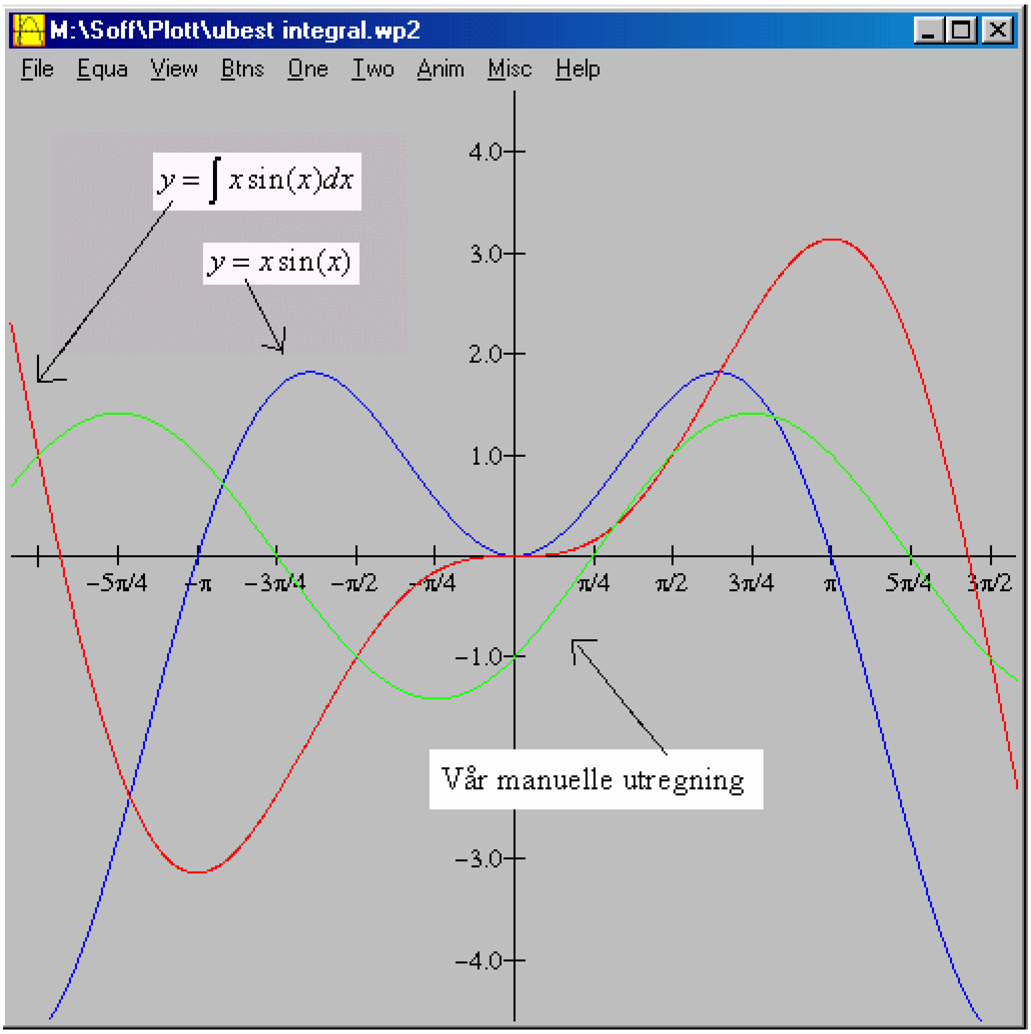


Integralkurven representerer en antiderivert av funksjonen $y = x\sin(x)$. En funksjon har et vilkårlig antall antideriverte, som skiller seg fra hverandre ved en konstant. Den antideriverte vi har plottet, er den som skjærer *x*-aksen i $x = 0$. Når vi ber *Winplot* om å tegne et ubestemt integral, angir "lower limit" hvor kurven skal skjære *x*-aksen. Veldig misvisende, men slik er det.

Prøv med forskjellige verdier i "lower limit" og se hvordan kurven flytter seg, men beholder formen.

Bruk av *Winplot* til å kontrollere egne utregninger

La oss si at vi har prøvd å beregne $\int x\sin(x)dx$ for hånd, og kommet fram til at svaret er $\sin(x) - \cos(x) + C$. Vi ber *Winplot* om å tegne inn denne funksjonen i plottet vi har fra før. Resultatet er vist under:



Det er opplagt at vi har regnet feil. Hadde vi regnet rett, skulle den grønne kurven - vår beregning, hatt samme form som den røde - *Winplots* beregning, muligens parallellforskjøvet. Den grønne kurven har en helt annen form. Vi går beregningene etter i sømmene, og finner ut at svaret skulle være $\sin(x) - x\cos(x) + C$. Vi editerer ligninga, og den grønne og røde kurven føyer seg pent sammen.

Vi kan ikke bruke *Winplot* til å bevise at vi har regnet riktig. Selv om vår og *Winplots* kurve skulle se like ut, behøver de ikke å være det. Muligens skiller de lag et sted utenfor plottet, muligens ligger de ikke helt oppå hverandre. Det vi kan bruke *Winplot* til, og som i seg selv er svært nyttig, er å påvise at vi har regnet feil.

Winplot - taylorpolynomer

Dette avsnittet tar for seg hvordan en tegner opp taylorpolynomet til en funksjon i et gitt punkt.

Winplot har mulighet for å tilnærme en kurve med et taylorpolynom. Ved å velge "One" - "Slider" i plottevinduet, får du opp slider-vinduet, som beskrevet i *Winplot kurver*.

I dette vinduet finnes en knapp som heter "Taylor approx". Klikker du på den, tegner *Winplot* opp kurven som representerer taylorpolynomet i det punktet trådkorset står. Du kan selv bestemme hvor mange ledd som skal tas med i taylorpolynomet, maksimalt ni. Antall ledd kalles ofte taylorpolynomets grad, og det tilhørende valget i slider-vinduet heter derfor "degree".

Taylorpolynomene kommer opp som innslag i inventarboksen, og du kan endre farge på dem, for å lettere å kunne holde kurvene fra hverandre.

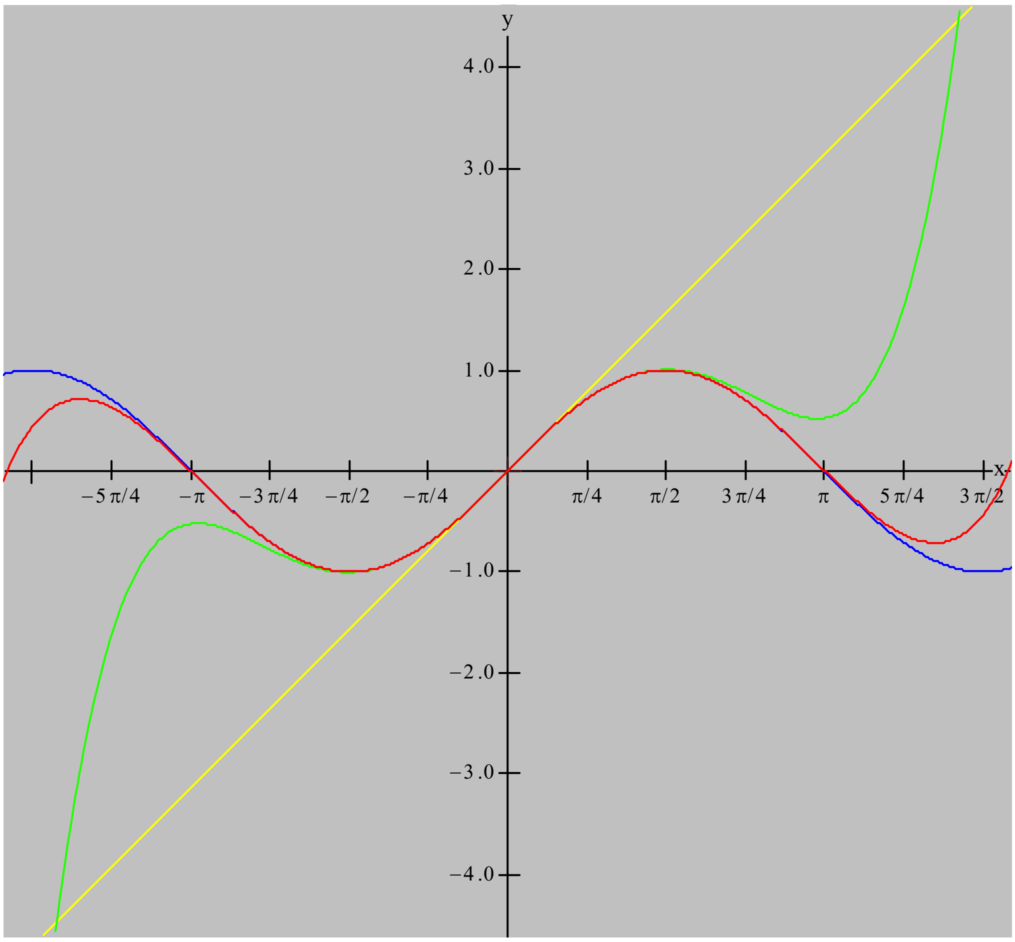
Som et eksempel skal vi plotte taylorpolynomet til $\sin(x)$ av grad 1, 5 og 9, omkring punktet $x = 0$.

Velg "Equa" - "Explicit..." og skriv "sin(x)". Velg fornuftige enheter på aksene, slik det er beskrevet i avsnittet "Plotting av trigonometriske funksjoner". Farg kurven blå.

Sett trådkorset i x = 0 på kurven til $\sin(x)$. Sett "degree" til 1, og klikk på "Taylor approx".

En kurve (rett linje) dukker opp i plottevinduet, og et nytt innslag kommer i inventarvinduet. Bruk "edit" til å farge kurven gul.

Gjør tilsvarende for grad 5 og 9, og farg kurvene henholdsvis grønn og rød. Resultatet blir som vist under:



Det er lett å se at tilnærmingen blir bedre jo høyere grad av taylorpolynomet en bruker.

Winplot - polarkoordinater

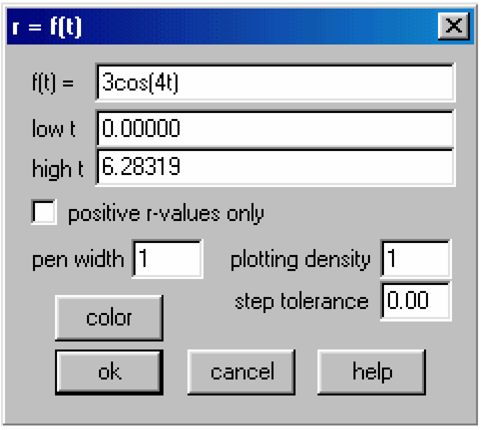
Dette avsnittet tar for seg tegning av kurver i polarkoordinater.

I de fleste veiledningene har vi plottet *y* som funksjon av *x* i et rettvinklet koordinatsystem. Men *Winplot* tilbyr også tre andre måter å plotte på. En av dem er polarkoordinater. I stedet for å plotte $y = f(x)$, plotter en $r = f(\theta)$, der *r* angir avstanden fra origo, og θ vinkelen linjen mellom *r* og origo danner med horisontalplanet. I *Winplot* skriver en *t* i stedet for θ, som er den greske bokstaven "theta".

Som eksempel skal vi plotte funksjonen $r = 3\cos(4\theta)$.

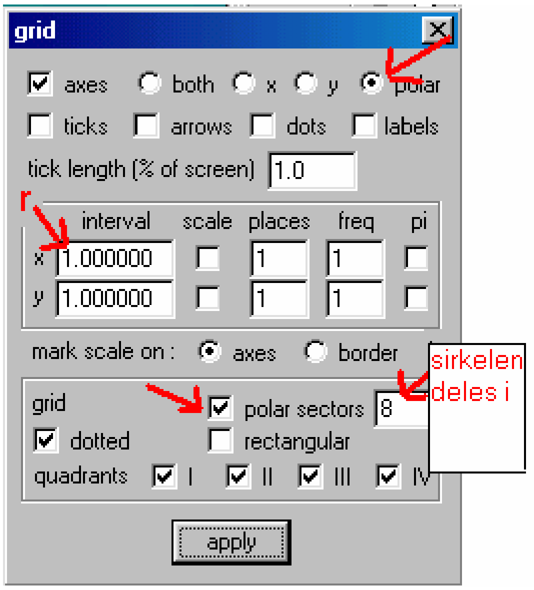
I plottevinduet starter vi som ellers med "Equa", men i stedet for "Explicit..." velger vi "Polar". Denne navngivingen er litt forvirrende. Plottet i polarkoordinater er også eksplisitt, så det burde hett "Explicit polar", eller noe slikt.

I vinduet vi får opp, skriver vi "3cos(4t)", som vist under:



Winplot foreslår at vi skal plotte for *t* i intervallet $[0,2\pi]$. Vi klikker "ok", og kurven dukker opp i plottevinduet.

Men i stedet for de vanlige x- og y-aksene ønsker vi å vise et polart koordinatsystem, der sirkler rundt origo representerer konstante verdier av *r*, og stråler ut fra origo representerer konstante verdier av *t*. Vi velger å vise *r* = 1, *r* = 2, etc. og t = $\frac{\pi}{4}$, dvs. $45°$, t = $\frac{\pi}{2}$, dvs. $90°$, etc. Vi åpner grid-vinduet og fyller ut som vist under:

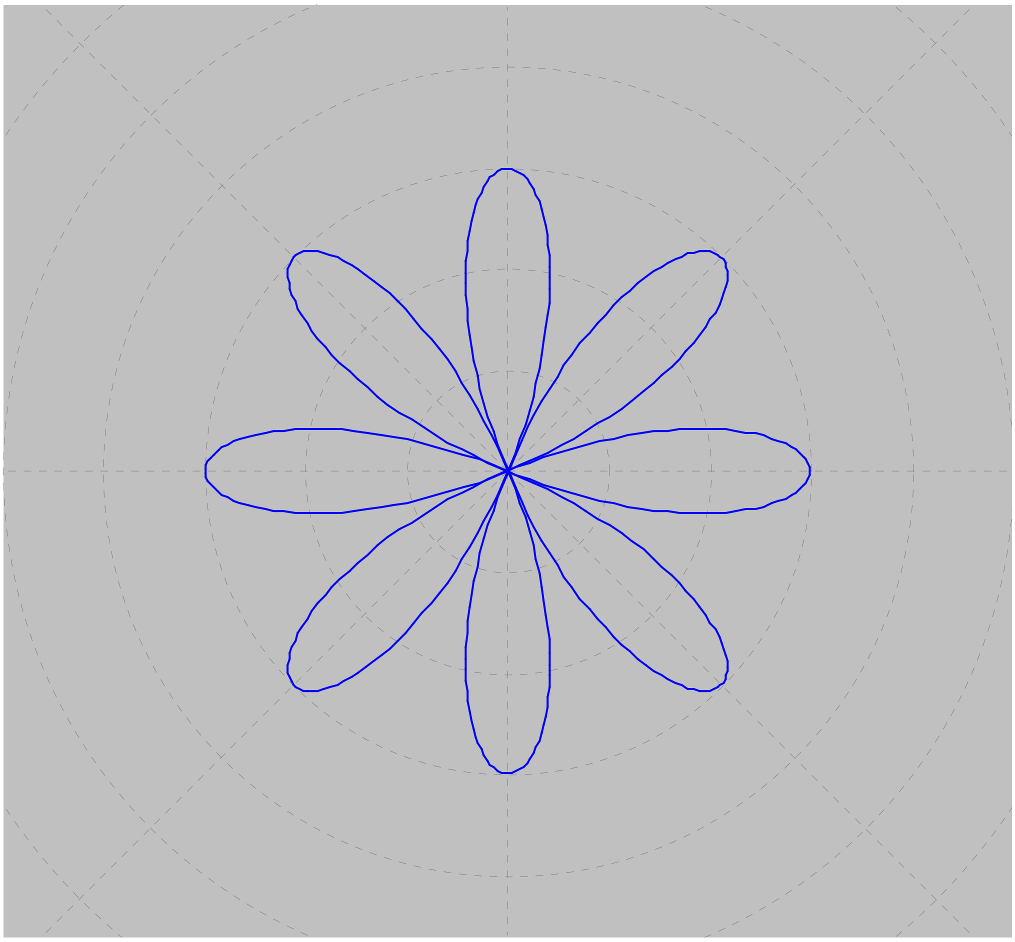


Vi krysser av for "polar" og "grid - polar sectors". *x* representerer i dette tilfellet *r*, og ved å sette *x* til 1, vil vi få sirkler med innbyrdes avstand 1. For å få tegnet opp vinklene $45°$, $90°$, etc., må vi dele sirkelen i 8.

Dessverre må det sies at grid-vinduet er inkonsistent og forvirrende for polarkoordinater.

Ikke fungerer det ordentlig heller. Endrer en verdien til x, gjenspeiles ikke endringen automatisk i plottevinduet. Men det pleier å hjelpe å skru av og på "scale".

I alle fall, etter å ha klikket på "apply", skal vi sitte igjen med dette:



Eksperimenter gjerne videre på egen hånd. Prøv for eksempel å bytte ut *4t* med *2t*. Hva skjer da med kurven?

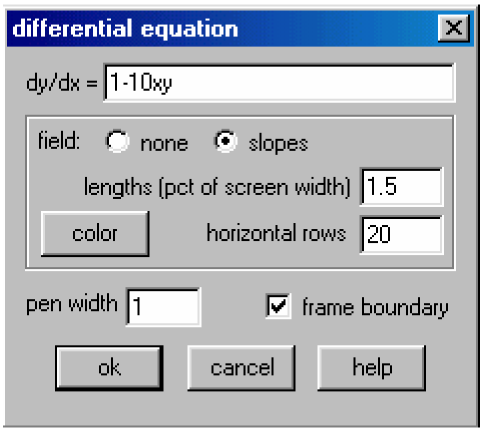
Winplot - differensialligninger

Dette avsnittet tar for seg bruk av *Winplot* i forbindelse med differensialligninger.

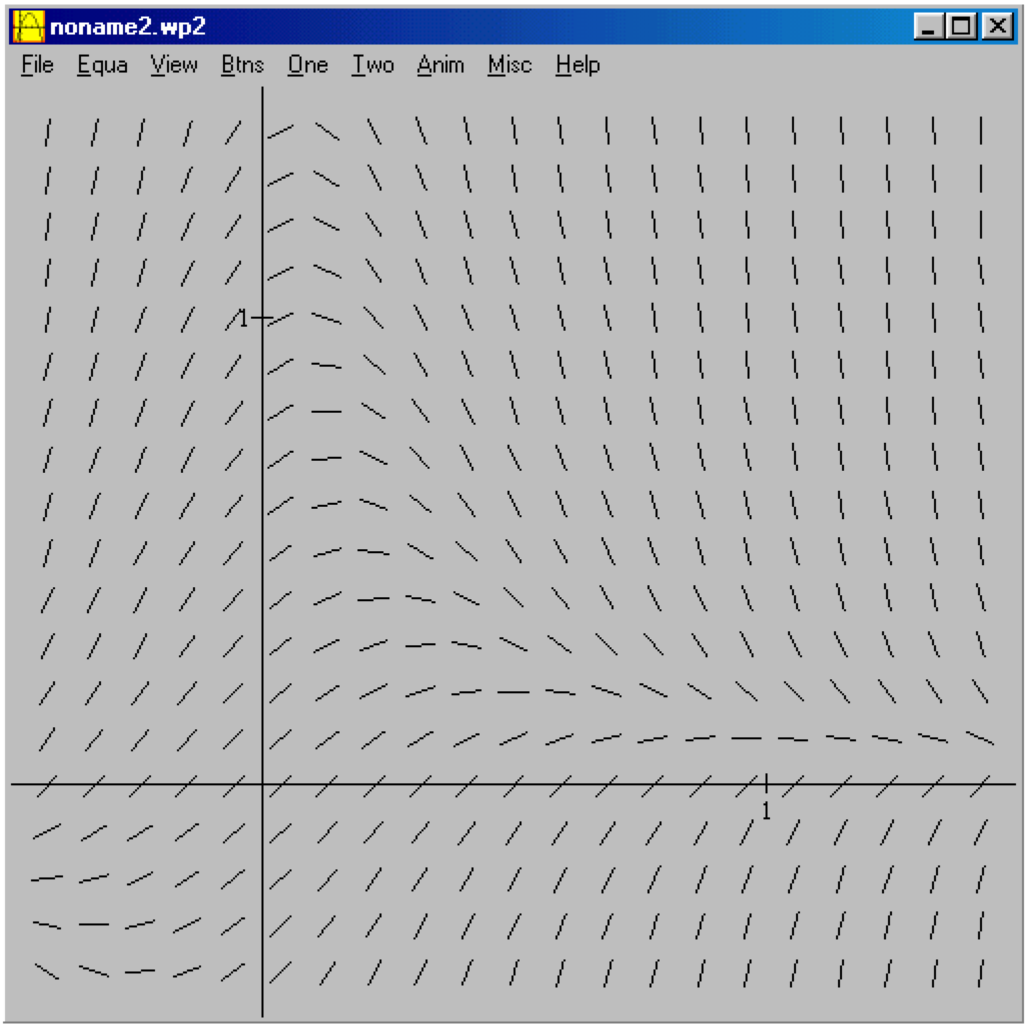
I *Winplot* finnes det vi har bruk for i forbindelse med differensialligninger på to plasser.

For å få fram et retningsdiagram, går vi på "Equa"-menyen. Der velger vi "differential" og "dy/dx...". Differensiallignings-vinduet kommer opp.

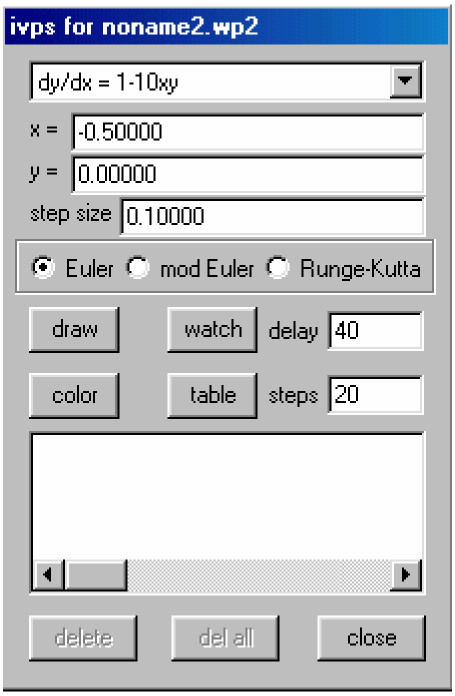
Vi vil studere differensialligningen $\frac{dy}{dx} = 1 - 10xy$, fyller ut differensiallignings-vinduet som vist under, og klikker "ok".



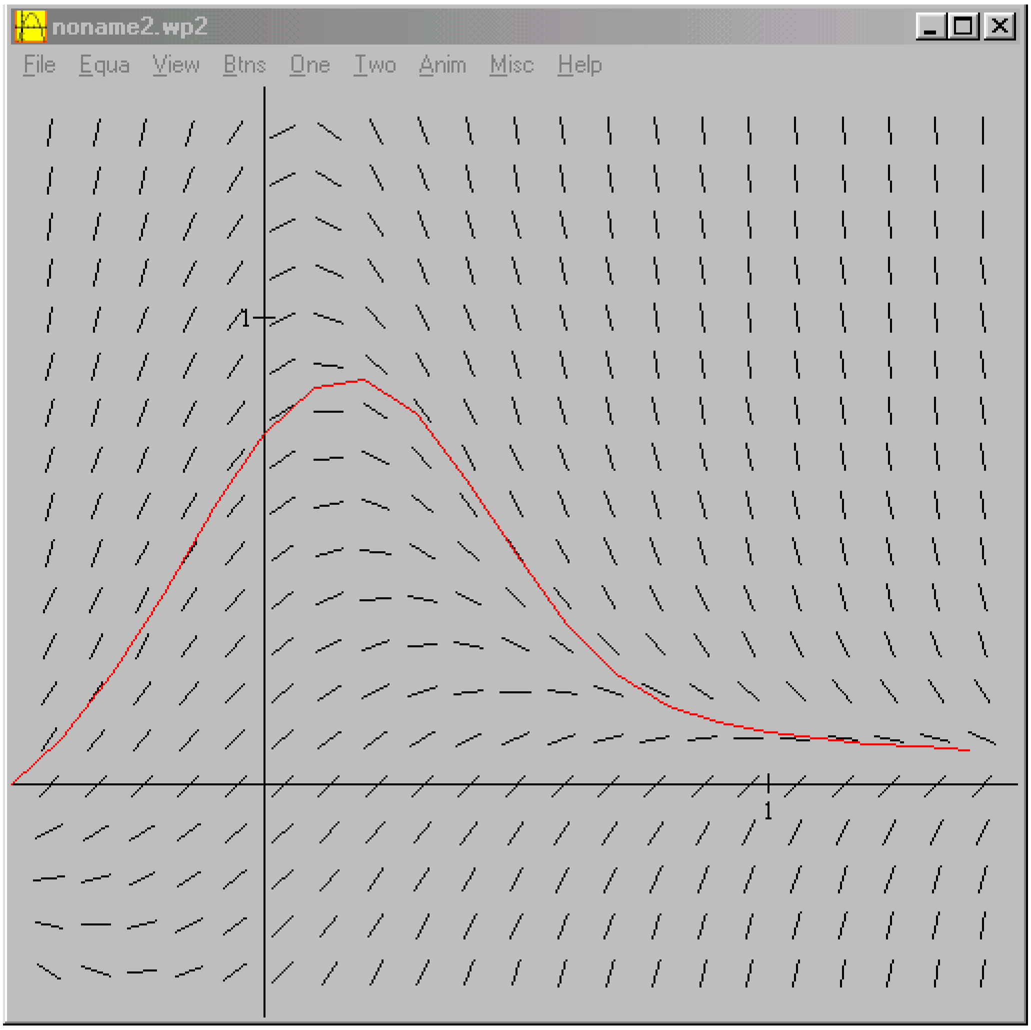
Etter at vi på "View"- "view…"-menyen har valgt $-0.5 \le x \le 1.5$, og $-0.5 \le y \le 1.5$, ser retningsdiagrammet slik ut:



For å tegne inn integralkurver, går vi til "One"-menyen og velger "dy/dx trajectory…". ivp-vinduet kommer opp. *ivp* er forkortelse for *Initial Value Problem*, og indikerer at integralkurvene er basert på en gitt startverdi. Vi bestemmer oss for å plotte en kurve med utgangspunkt i $(-0.5,0)$, og fyller ut *ivp*-vinduet som vist under:



Vi klikker på "draw", og en integralkurve tegnes inn:



Det er mulig å tegne så mange integralkurver en vil, ved å velge forskjellig startpunkter og klikke på "draw". Det er mulig å få en liste med punkter som integralkurven går igjennom ved å klikke på "table".

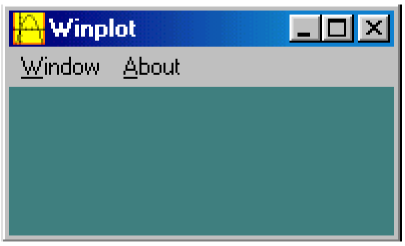
Ved å gjøre "step size" mindre, vil en få et mer nøyaktig plott, men det vil ta lengre tid å tegne opp kurven. "Euler", "mod Euler" og "Runge-Kutta" angir hvilken metode *Winplot* skal bruke for å beregne hvor integralkurven skal gå, men det bryr vi oss ikke med å forklare nærmere.

Winplot 3d - intro

Dette avsnittet gir en introduksjon til hvordan en lager tredimensjonale plott. Det vil være en fordel å ha arbeidet noe med todimensjonale plott før du tar fatt på dette avsnittet.

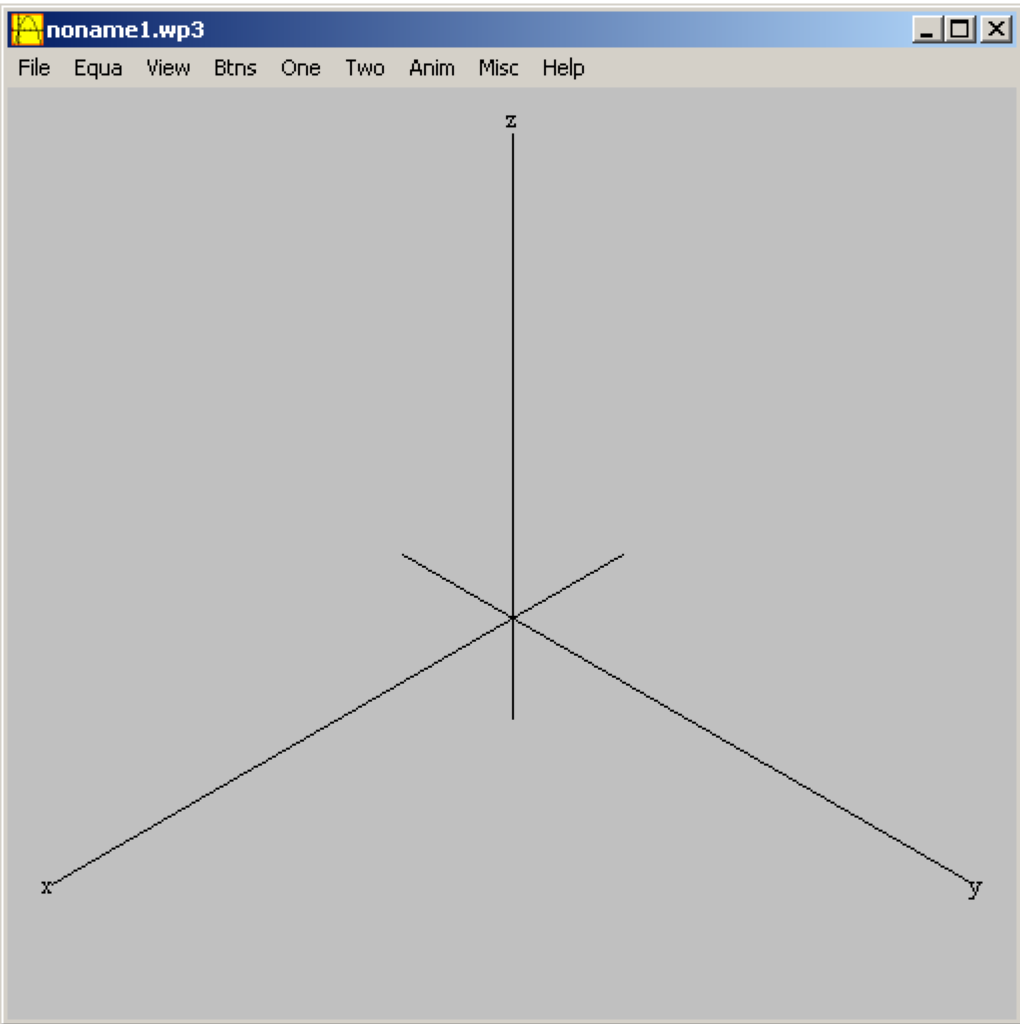
Åpne et 3d plottevindu

Når *Winplot* starter, kommer vinduet under opp, slik som beskrevet i "Winplot intro".



Men i stedet for å gå videre med "Window" - "2-dim", som tidligere, velger vi nå "Window" - "3-dim".

Plottevinduet vist under kommer opp. Til forskjell fra det todimensjonale plottevinduet, inneholder dette tre akser. I stedet for å plotte *y*-verdier som en funksjon av *x*, skal vi nå plotte *z*-verdier som en funksjon av både *x* og *y*.



En liten advarsel

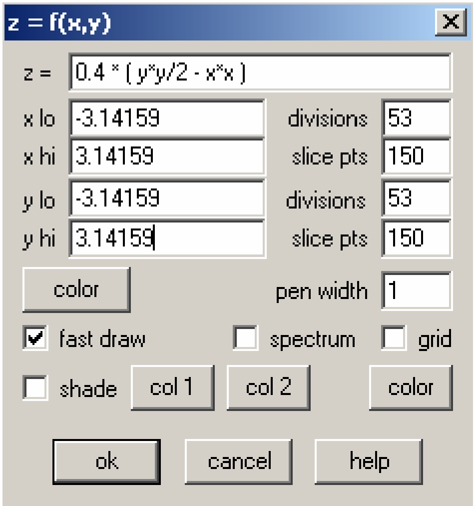
Winplot er dessverre ikke alltid like medgjørlig når det gjelder tredimensjonale plott, og det er fort å gå i baret. Noe er direkte feil, og andre problemer skyldes at *Winplot* velger uklokt når den setter diverse parametere. Men samtidig er *Winplot* i stand til å gi flotte resultater når den bare blir holdt i ørene. Eksperimenter derfor gjerne på egen hånd, men bli ikke skuffet hvis noe ikke ser ut som du har tenkt. Det er ikke sikkert at feilen er din.

Lage et plott

Vi skal plotte en funksjon som gir en flate som ligner en sadel: $\frac{y^2}{2} - \frac{x^2}{1} = \frac{z}{0.4}$.

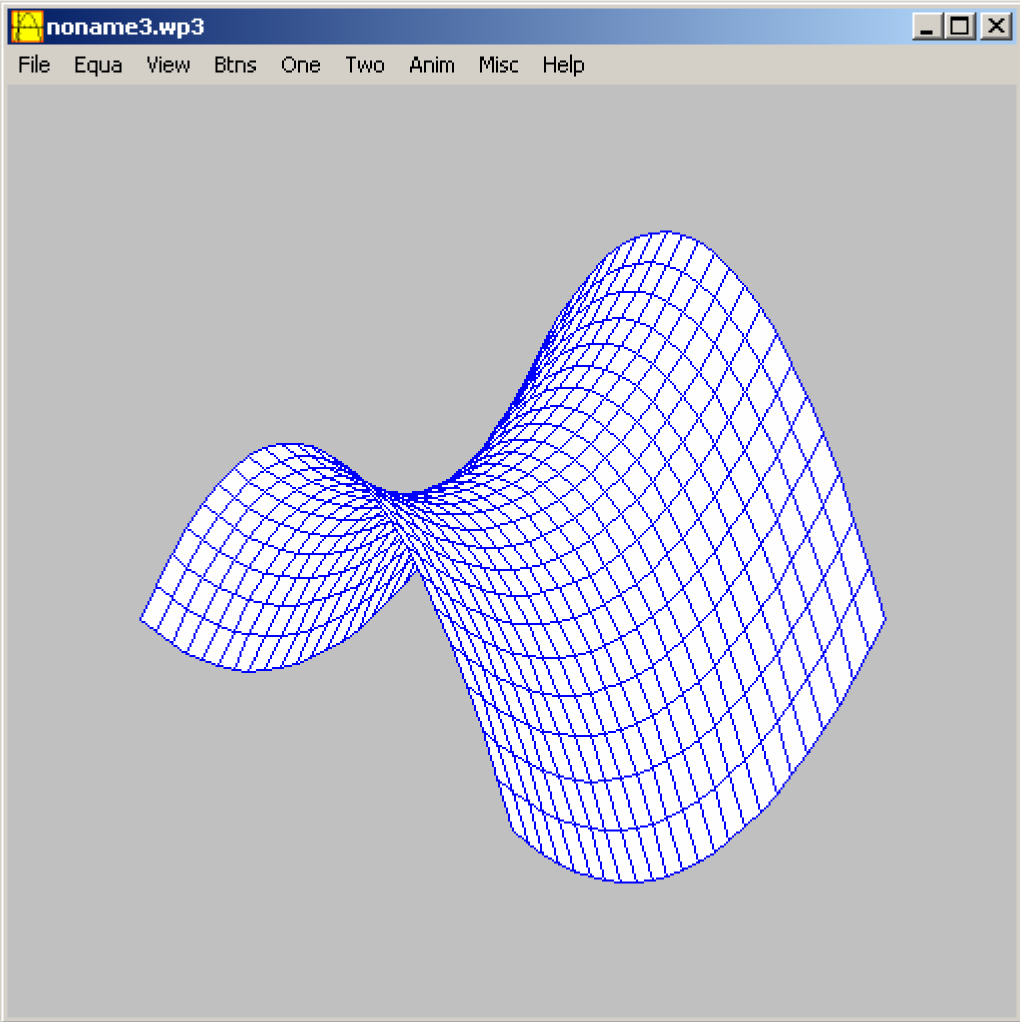
Velg "Equa" - "Explicit…". i plottevinduet Dette gir oss muligheten til å plotte z som en funksjon av x og y.

Vi får opp en dialogboks der vi fyller ut "z=" med "0.4 * (y*y / 2 - x*x)", som vist under:



Pass på at det er krysset av for "fast draw".

Det resulterende plottet ser noe slik ut:



Det er ikke sikkert ditt plott ble helt likt, det kan være det ligger orientert annerledes. Vi skal snart se hvordan vi endrer dette.

Skru av og på akser

Ved å velge "View" - "Axes" - "Axes", kan en skru av og på visning av de tre aksene. Av en eller annen grunn skrudde *Winplot* av aksene da vi laget plottet over. Skru dem på igjen, så er det lettere å se hva som er opp og ned.

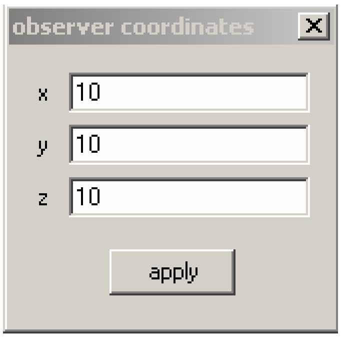
Valget "Solid arrowheads", som burde gitt piler på aksene, ser ikke ut til å virke.

Zoom og rotasjon

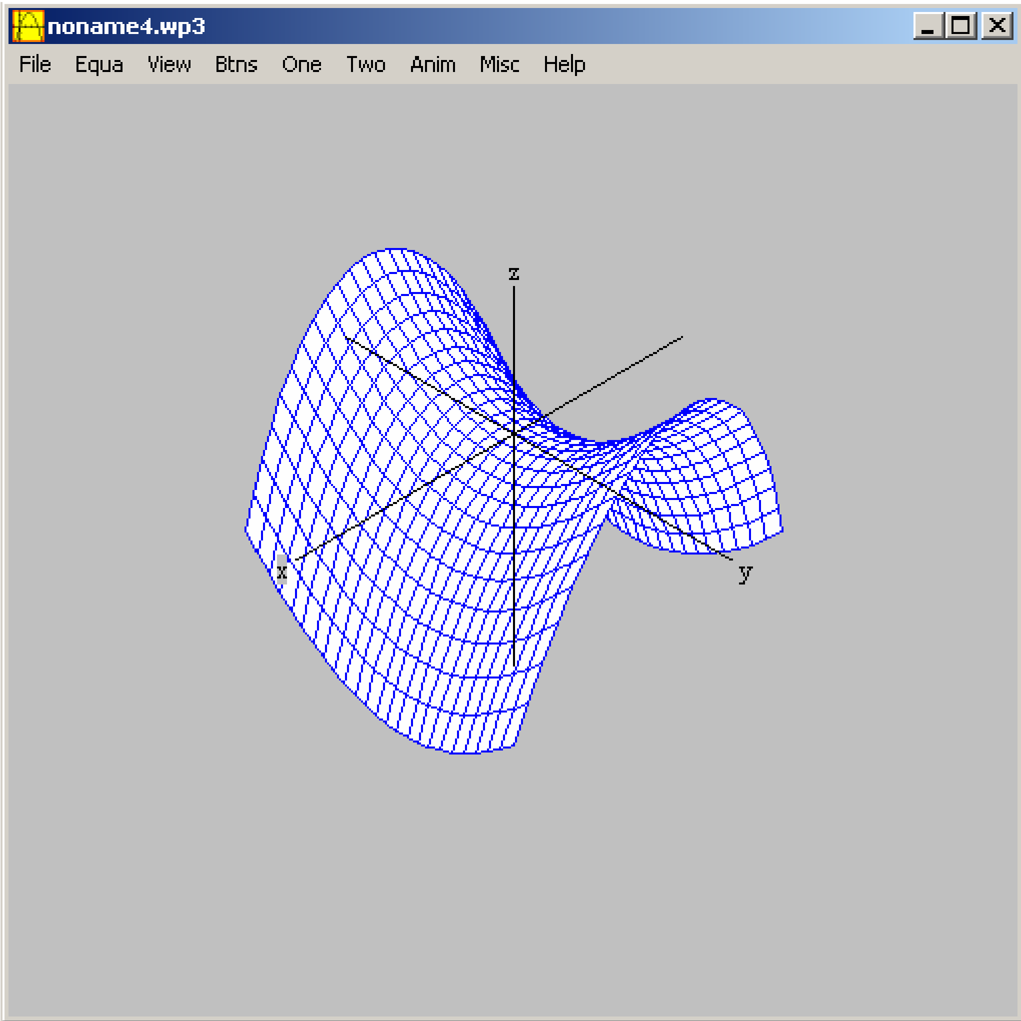
Et tredimensjonalt plott kan zoomes inn og ut og roteres ved å bruke tastaturet, så vel som menyene. *PageUp* og *PageDown* zoomer, mens piltastene roterer. Pass på å gjøre plottevinduet aktivt ved å klikke i det, før du prøver dette.

Endring av betraktningspunkt

Ved å velge "View" - "Observer" - "Coordinates…", kan du velge hvor i rommet du vil stå og betrakte flaten som er plottet. Skriv inn koordinatene (10, 10, 10), som vist under, og trykk "apply".



Plottet skal nå se omlag slik ut:



Prøv litt forskjellige koordinater, og studer hvordan det påvirker det bildet du ser av den plottede flaten.

Underlig nok synes ikke avstanden til flaten å ha noe å si for hvor stor den ser ut. En får det samme resultatet ved å sette koordinatene til (100, 100, 100) som til (10, 10, 10).

Rotasjon og betraktningspunkt

La vinduet "observer coordinates" stå framme, samtidig som du roterer flaten ved hjelp av piltastene. Verdiene i "observer coordinates" endrer seg! Med utgangspunkt i (10, 10, 10), vil et trykk på høyrepil først gi (-10, 10, 10), så

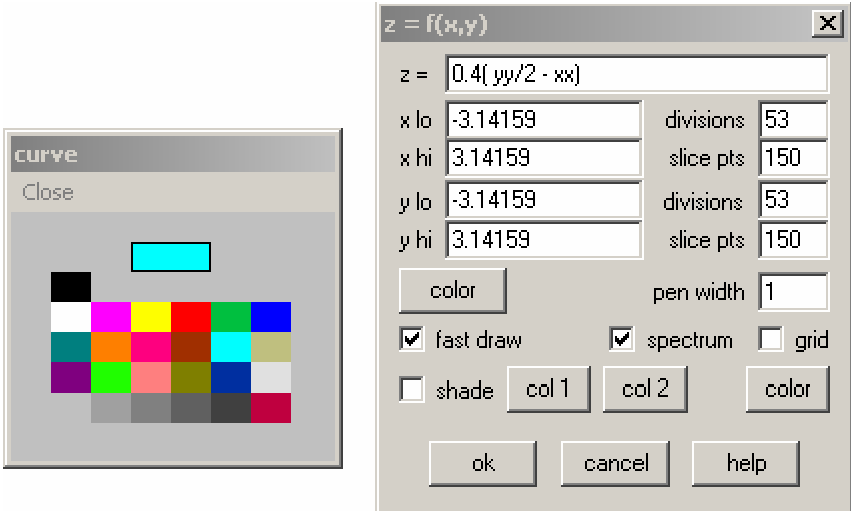
(-10, -10, 10), (10, -10, 10), og tilbake til (10, 10, 10). Pil opp gir økende *z*-verdier, mens *x*- og *y*-verdiene holdes konstante. Tilsvarende gir pil ned avtagende *z*-verdier.

En rotasjon av en flate i *Winplot* utføres altså ved at betrakteren flyttes rundt i rommet, og ser flaten fra forskjellige vinkler.

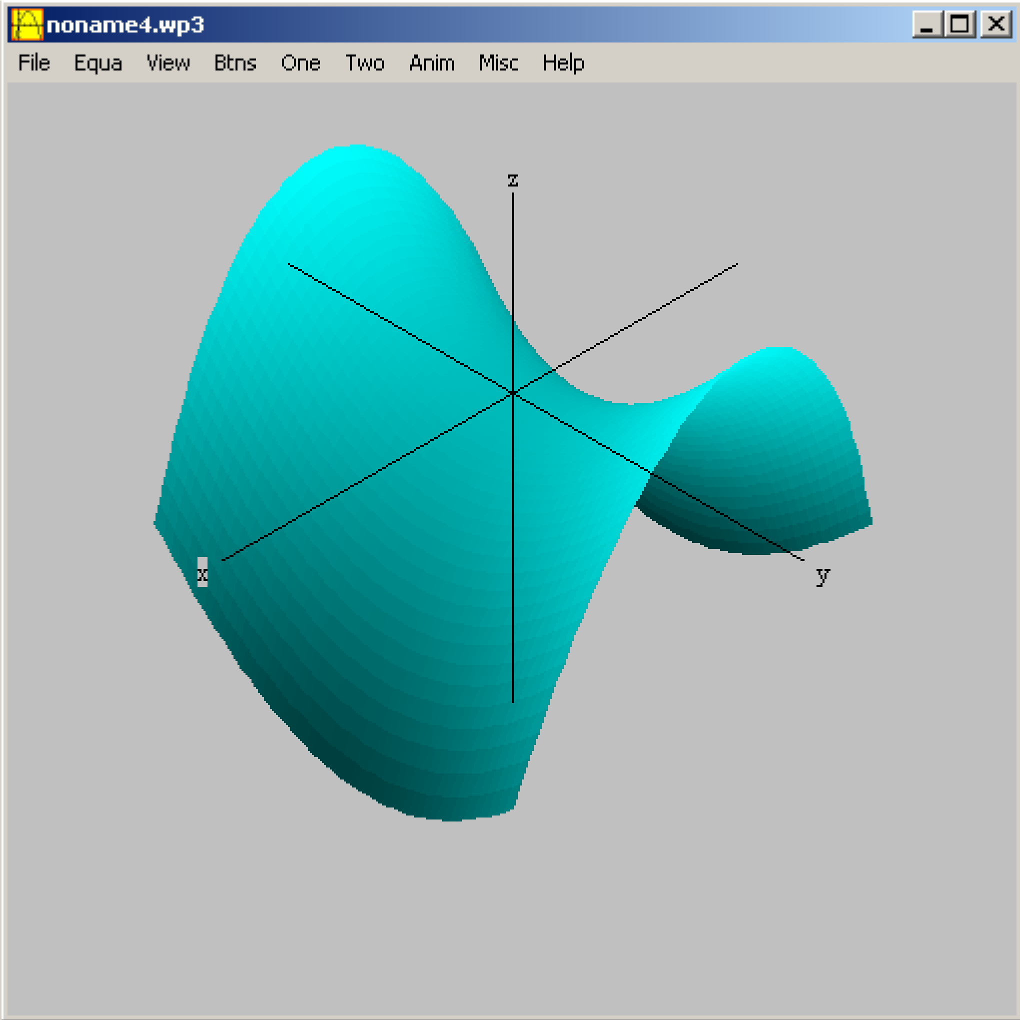
Skyggelegging

Sett "observer coordinates" til (10, 10, 10), og zoom gjerne litt inn for å få figuren større. Velg "edit" i inventarvinduet. Hvis inventarvinduet ikke er framme, henter du det ved å velge "Equa" - "Inventory…".

Ved å krysse av for "spectrum", kan du gi flaten en fin skyggeeffekt. Det er viktig å velge en lys farge til dette. Klikk på "color". Ikke rør "col1" eller "col2", de brukes til noe annet, som beskrevet i neste avsnitt, "Fargelegging". Velg for eksempel lyseblå, og klikk "ok". Verdiene i feltene "divisions" kan med fordel økes litt, for å en glattere flate. 53 ser ut til å være det maksimalt tillatte.



Det resulterende plottet skal se omlag slik ut:



Slett ikke verst!

Fargelegging

I stedet for å bruke "spectrum", kan du krysse av for "shade". Du vil da kunne gi over- og undersiden av en flate hver sin farge. Velg først "color" og sett denne til svart. Så velger du etter tur "col1" og "col2" og gir dem forskjellige farger, for eksempel rød og blå. Disse fargene behøver ikke være spesielt lyse.

Parametrisering

Flaten vi har plottet, er en såkalt hyperbolsk paraboloide, med den generelle formelen

$\frac{y^2}{b^2} - \frac{x^2}{a^2} = \frac{z}{c}$. I vårt plott har vi brukt $a^2 = 1, b^2 = 2, c = 0.4$.

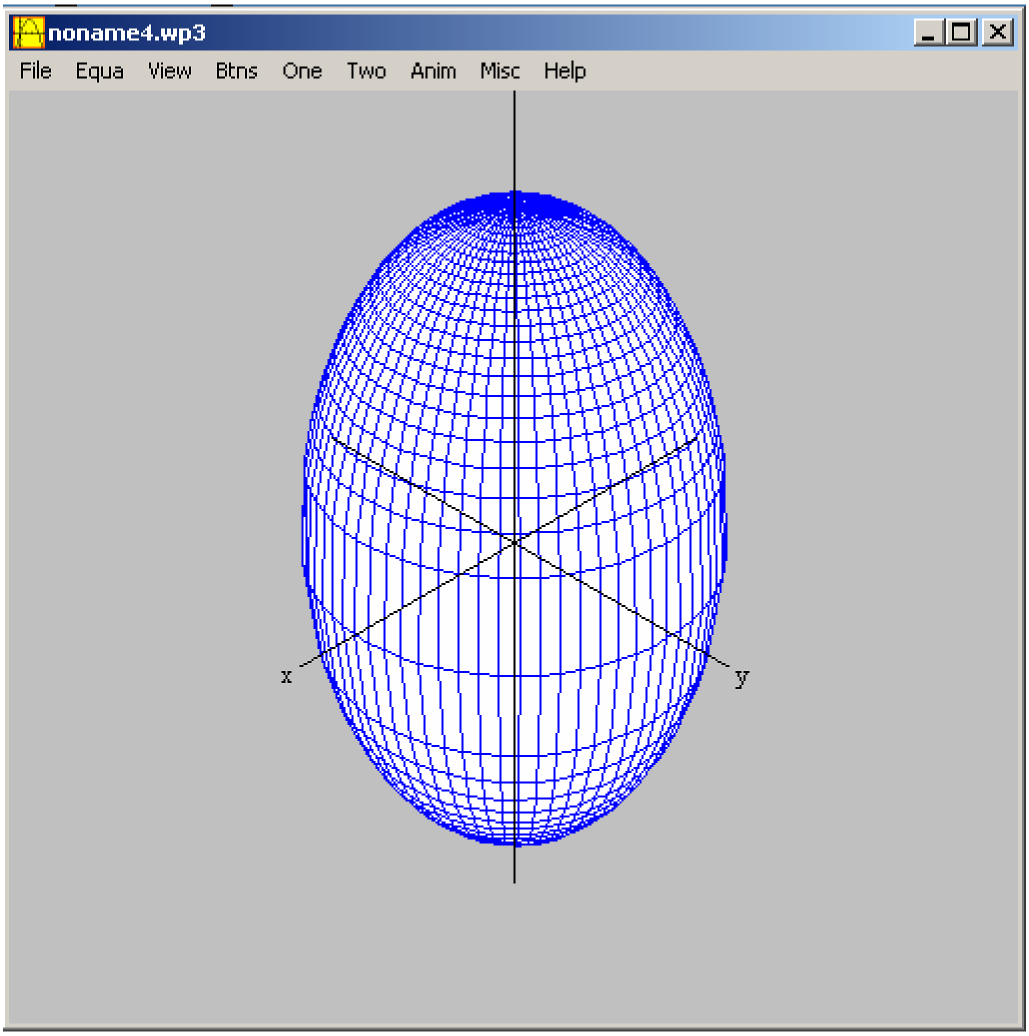
I avsnittet "Endre parametere med blafelt" beskrives det hvordan en kan legge inn parametere i formler, endre dem dynamisk med blafelt, og se hvordan kurven da endrer seg. Tilsvarende burde vært mulig for tredimensjonale plott også, slik at vi kunne lagt inn *a*, *b* og *c* og studert hvordan flaten endret seg når de varierte, men det ser dessverre ut til å fungere dårlig.

Winplot 3d - problemer

Dette avsnittet diskuteres noen problemer som kan oppstå når en lager tredimensjonale plott. Det vil være en fordel å ha arbeidet noe med både to- og tredimensjonale plott før du tar fatt på dette avsnittet. Du må også være fortrolig med bruk av polarkoordinater.

Vårt mål

Vi setter oss fore å lag et plott av ellipsoiden $\frac{x^2}{1}+\frac{y^2}{1}+\frac{z^2}{3}=1$, og håper å ende opp med noe som ligner på figuren under:

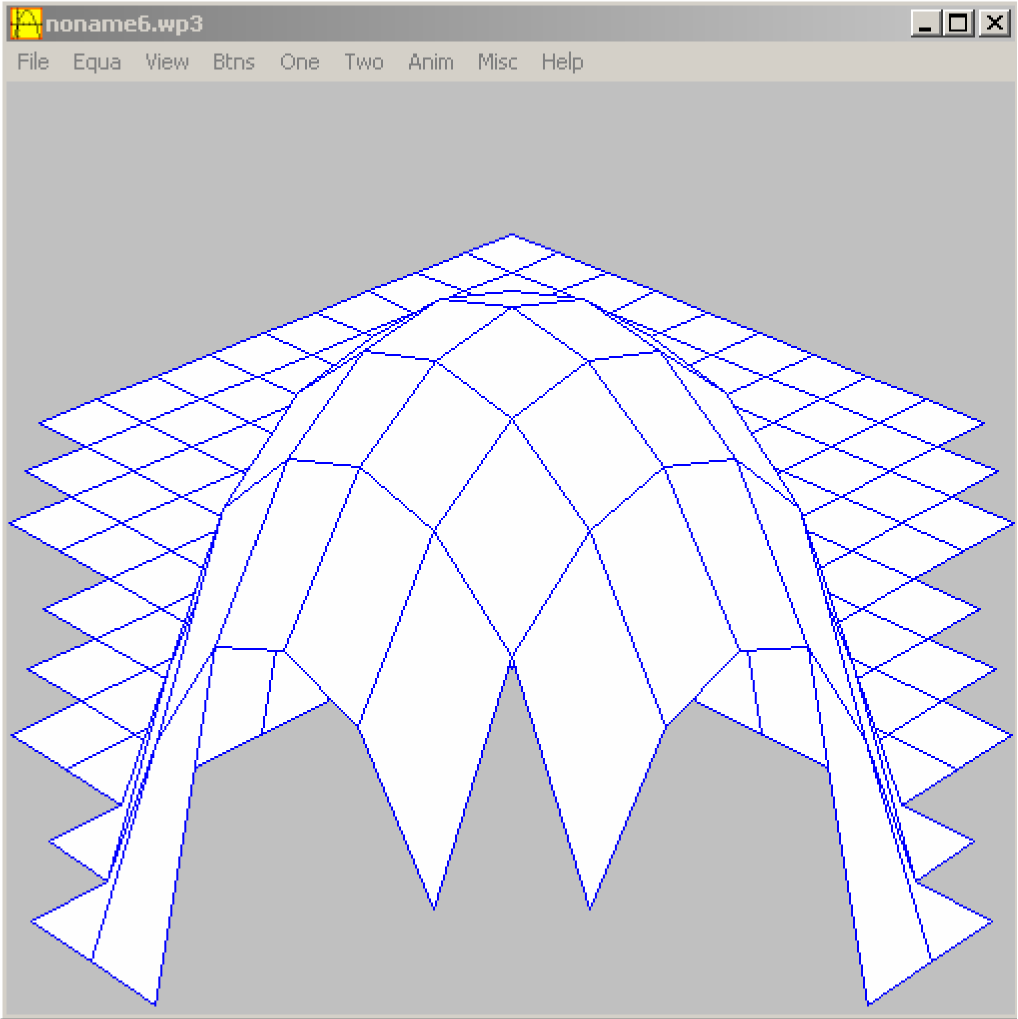


Første forsøk

For å kunne fôre *Winplot* med denne ligningen, løser vi den først med hensyn på z:

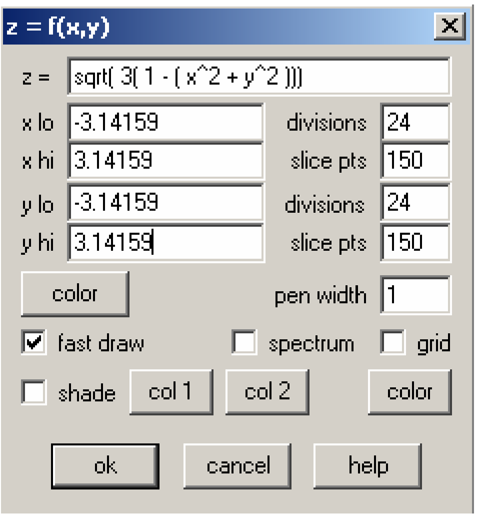
$$z=\pm\sqrt{3(1-(x^2+y^2))}$$

Dette blir to porsjoner til *Winplot*, en for positive z-verdier, og en for negative. Vi begynner med den positive, og håper at det skal resultere i et plott av den øvre delen av ellipsoiden. Men resultatet blir slik:



Hva gikk galt?

Ser vi på boksen der vi skrev inn funksjonen, oppdager vi at *Winplot* har foreslått et definisjonsområde for både x og y på $[-\pi, \pi]$:



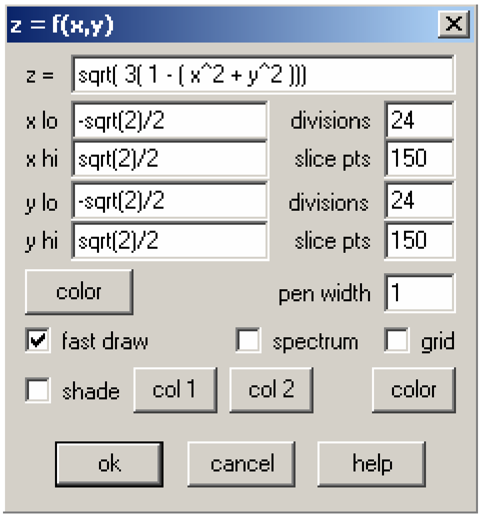
Men mange kombinasjoner av verdier i dette intervallet fører til at vi får negative verdier under rottegnet, for eksempel gir $x = 1, y = 1$ at $z = \sqrt{-3}$. *Winplot* klarer ikke å håndtere dette, og resultatet blir bare graps.

Andre forsøk

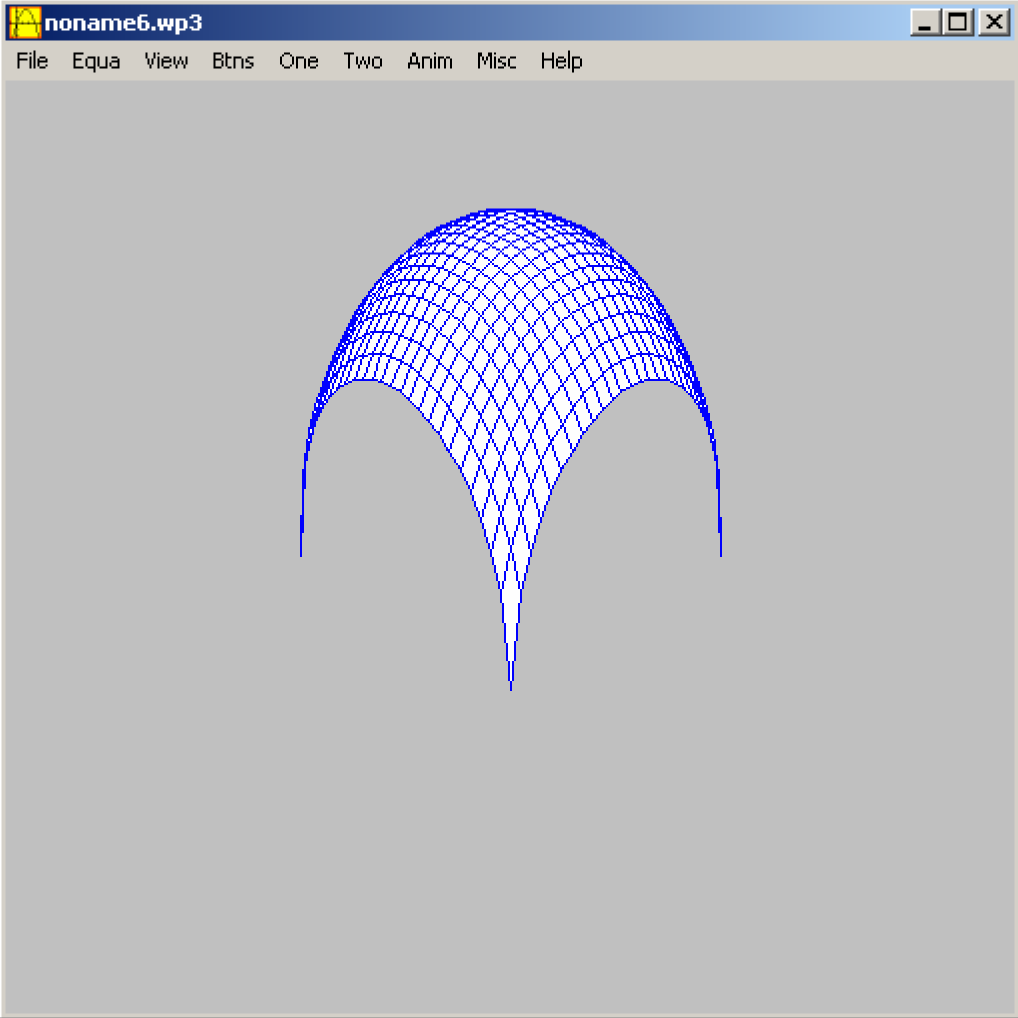
Vi vil prøve å begrense verdiene til *x* og *y* slik at vi ikke får negative tall under rottegnet. Vi må da ha $3(1-(x^2+y^2) \geq 0$, dvs. $x^2+y^2 \leq 1$.

Kravet blir altså at alle punktene (*x*, *y*) må ligge innenfor en sirkel med radius 1.

Men dette kan vi ikke fortelle *Winplot*, for *Winplot* vil ha faste grenseverdier for *x* og *y*. Vi kan ikke si at "grensene for *x* avhenger av verdiene til *y*". Det vi kan gjøre, er å prøve en trygg variant, der vi setter verdiområdene for både *x* og *y* til $[-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]$. Da vil den høyest tenkelige verdien til x^2+y^2 bli $\left(\frac{\sqrt{2}}{2}\right)^2 + \left(\frac{\sqrt{2}}{2}\right)^2 = \frac{2}{4}+\frac{2}{4} = 1$, og det vil ikke forekomme negative tall under rottegnet:



Nå ser resultatet slik ut:



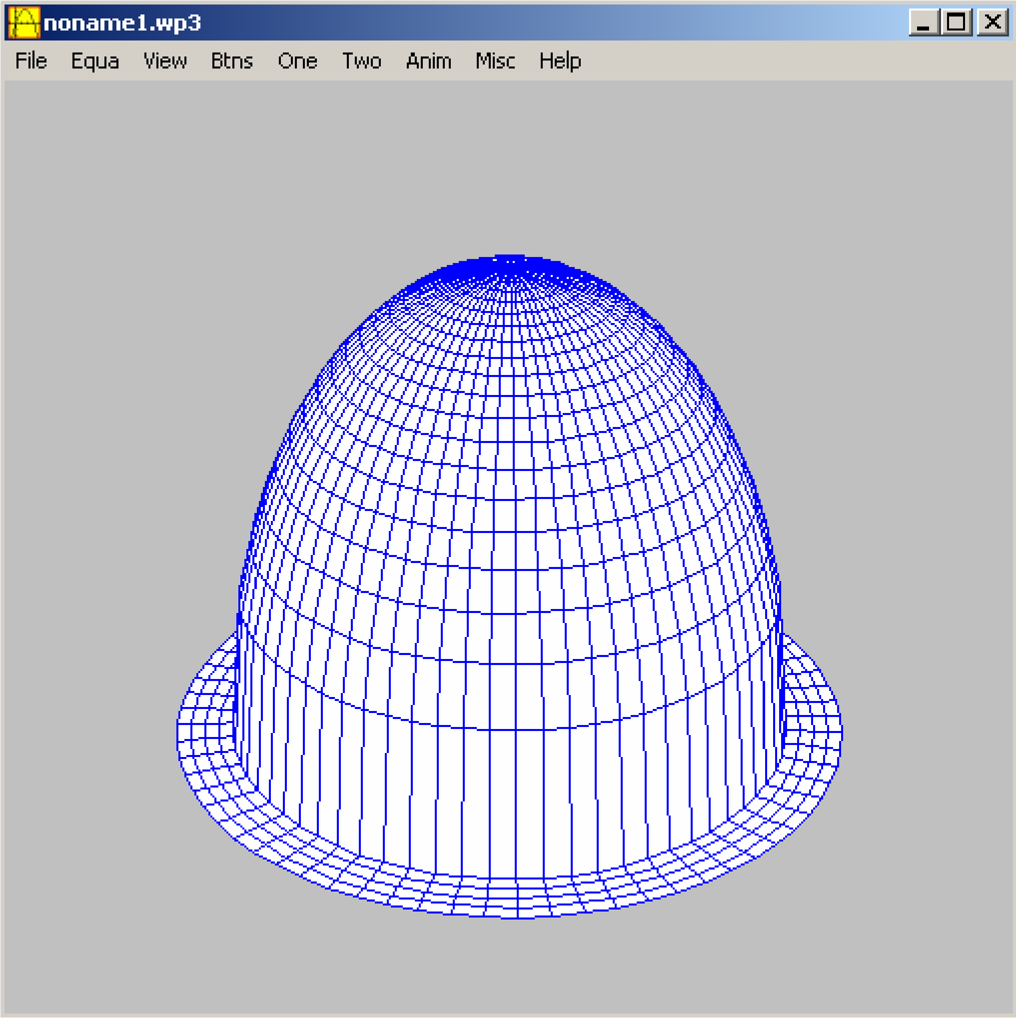
Ikke helt bra det heller. Formen ser riktig ut, men deler av figuren er skåret bort. Det var jo også det vi kunne vente når vi tok bort en del helt legitime kombinasjoner av *x*- og *y*-verdier fra definisjonsområdet.

Tredje forsøk

Det er ingen vei utenom, vi må skifte koordinatsystem. Det går ikke å oppgi fornuftige grenseverdier for x og y. I stedet for å operere med koordinatene (x, y, z) skifter vi til (r, θ, z). Dette er sylinderkoordinater, der *x* og *y* er skiftet ut med polarkoordinatene *r* og θ, og *z* er beholdt uendret.

Vi har at $r^2 = x^2 + y^2$. Ligningen for vår ellipsoide, $z = \pm\sqrt{3(1-(x^2 + y^2))}$, blir derfor $z = \pm\sqrt{3(1-r^2)}$ i sylinderkoordinater.

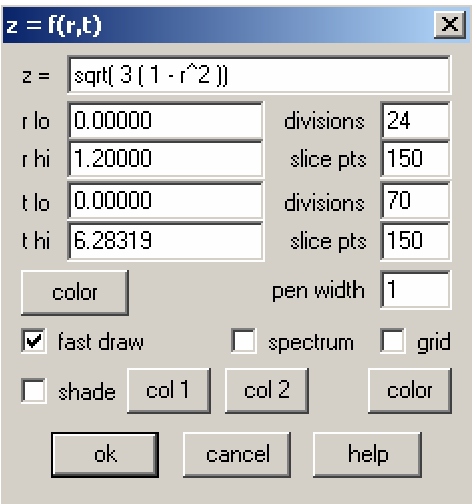
Vi åpner et nytt 3d plottevindu i *Winplot*, og velger "Equa" - "Cylindrical…", og skriver inn sqrt (3 (1 - r^2)). Nå ender vi opp med dette:



Fremdeles ikke helt bra, men vi er på rett spor.

Fjerde forsøk

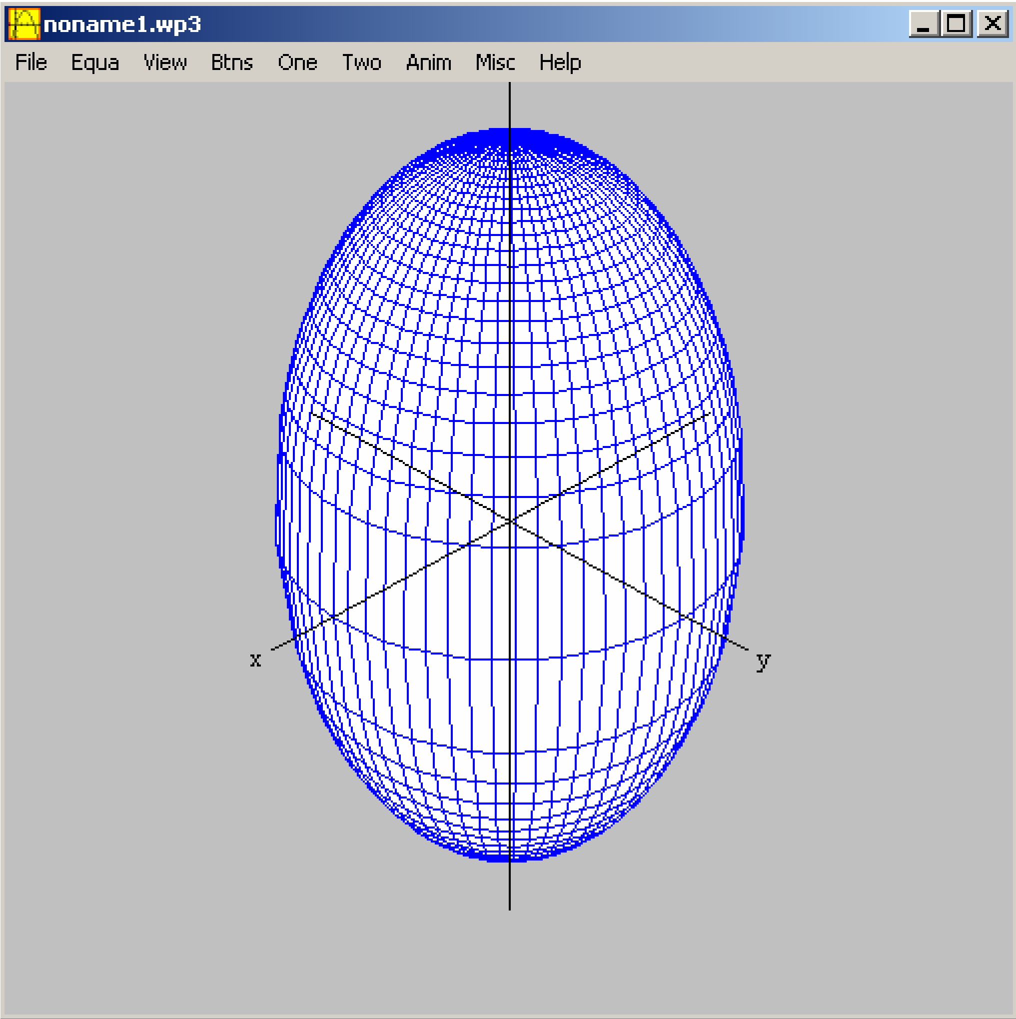
Vi tar en titt på hva Winplot har foreslått som grenser for r og θ:



Verdiene for θ, som *Winplot* kaller *t*, ser bra ut, de representerer en full sirkel. Men verdiene for r går for høyt, og fører til at vi får negative verdier under rottegnet, som før. Hvis vi erstatter "r hi " med 1, ser alt mye bedre ut.

Vi skrur på aksene, og registrerer at *Winplot* fortsetter å vise aksene i et kartesisk koordinatsystem, selv om vi har skiftet til polarkoordinater.

Nå har vi et flott plott av halvparten av ellipsoiden, $z = \sqrt{3(1-r^2)}$ Vi legger inn et plott av den andre halvparten, $z = -\sqrt{3(1-r^2)}$. Det gjør vi enklest ved å bruke "dupl" i inventarvinduet, og editere formelen. Vi zoomer litt, og ender opp med dette:



Som var det vi håpet på.