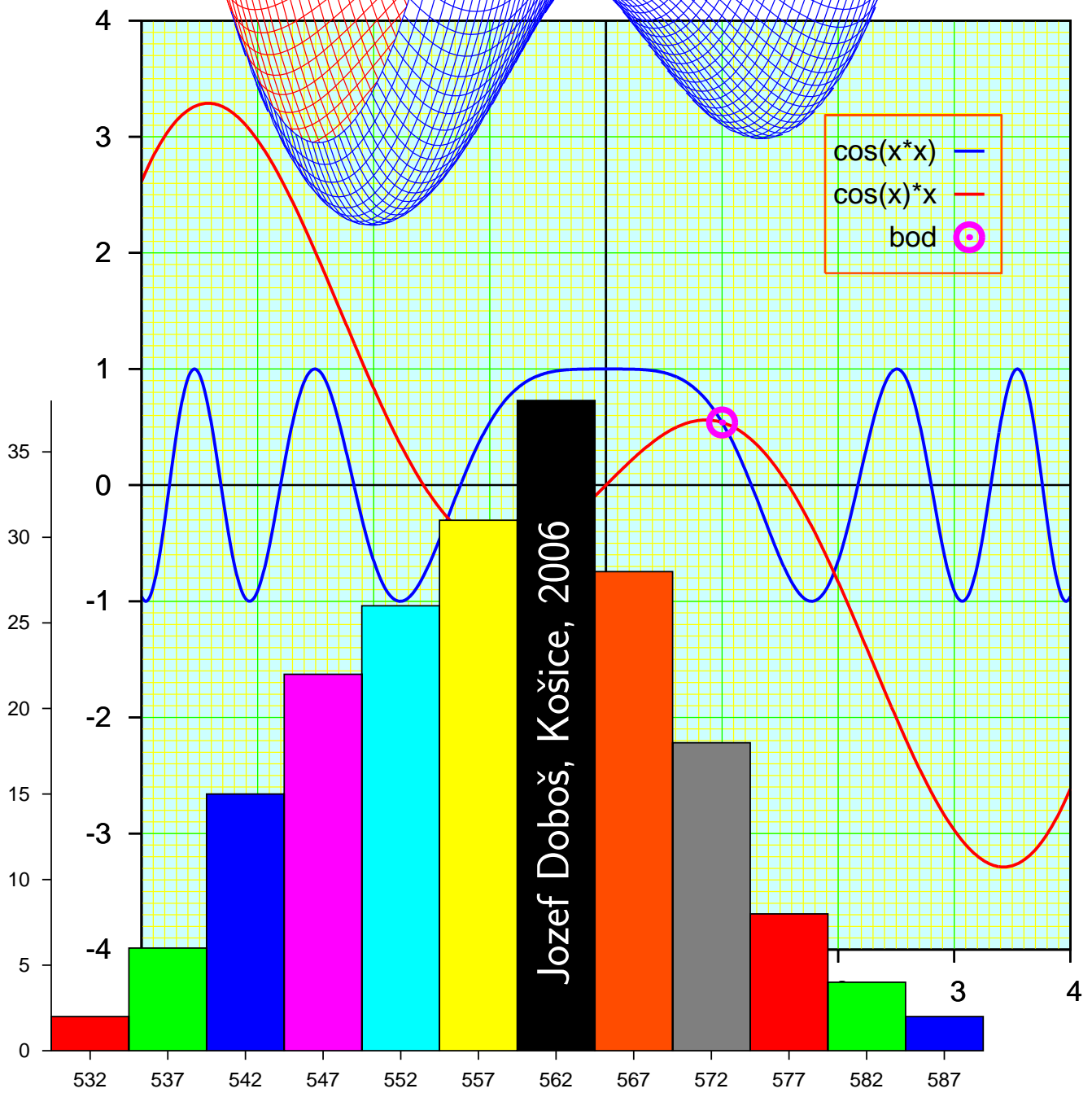
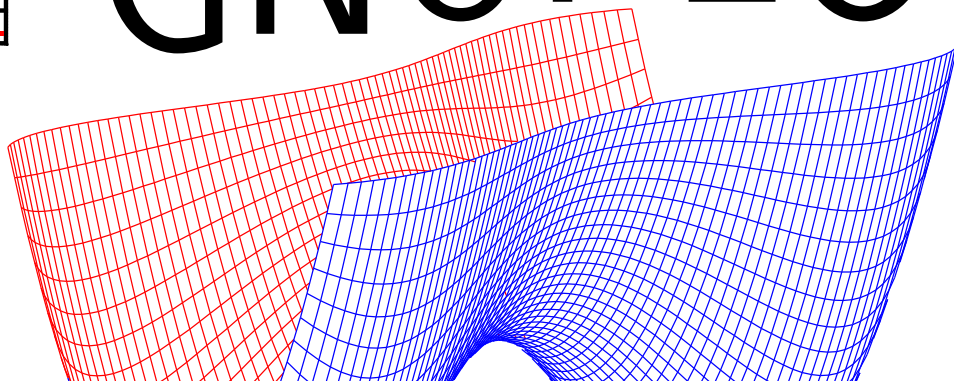


GNU PLOT



Táto publikácia vznikla s príspevím grantovej agentúry SR KEGA v tematickej oblasti Nové technológie vo výučbe – projekt: 3/2158/04 – Využitie OpenSource softvéru vo výučbe na vysokých školách.

Recenzovali: Ján Buša, Ladislav Ševčovič.

ISBN 80-8073-637-5

Sadzba programom $\mathcal{A}\mathcal{M}\mathcal{S}\mathcal{T}\mathcal{E}\mathcal{X}$

© Jozef Doboš, 2006.

Ktokoľvek má dovoľenie vyhotoviť alebo distribuovať doslovný opis tohoto dokumentu alebo jeho časti akýmkoľvek médium za predpokladu, že bude zachované oznámenie o copyrighte a oznámenie o povolení, a že distribútor príjemcovi poskytne povolenie na ďalšie šírenie, a to v rovnakej podobe, akú má toto oznámenie.

OBSAH

Začíname	4
Dátové súbory	7
Export grafu	13
Kozmetické úpravy	14
Legenda	14
Body a čiary	18
Rámček	19
Grafy funkcií	22
Grafy funkcií daných explicitne	26
Krivky dané parametricky	31
Polárne súradnice	33
Krivka daná implicitne	34
Plochy	35
Krivky v priestore	42
Čo je nové vo verzii 4.0	47
Animácie	53
Záver	54

Stručne povedané, **gnuplot** je interaktívny kresliaci program. Umožňuje nám vykresľovať bodové, spojnicové a stĺpcové grafy, dvojrozmerné a trojrozmerné grafy zo zadaných hodnôt (napr. výsledky merania), ako aj krivky a plochy zadané matematicky. Pritom **gnuplot** má podporu pre množstvo operačných systémov, grafických výstupných formátov a výstupných zariadení, umožňuje robiť výpočty v reálnej a komplexnej aritmetike.

Inštalčné súbory, ako aj ďalšie informácie o programe **gnuplot** nájdete na adrese

<http://www.gnuplot.info/>


V tejto brožúrke sa budeme zaoberať verziou 4.0 pre Windows. Všetko potrebné pre inštaláciu je v súbore `gp400win32.zip`. Stačí tento súbor rozbaľiť do vhodného adresára (napr. Program Files). Na spustenie programu slúži súbor `wgnuplot.exe`. Ihneď po spustení programu môžete kresliť grafy. K dispozícii máte interaktívny help, ako aj 120 stranový manuál.

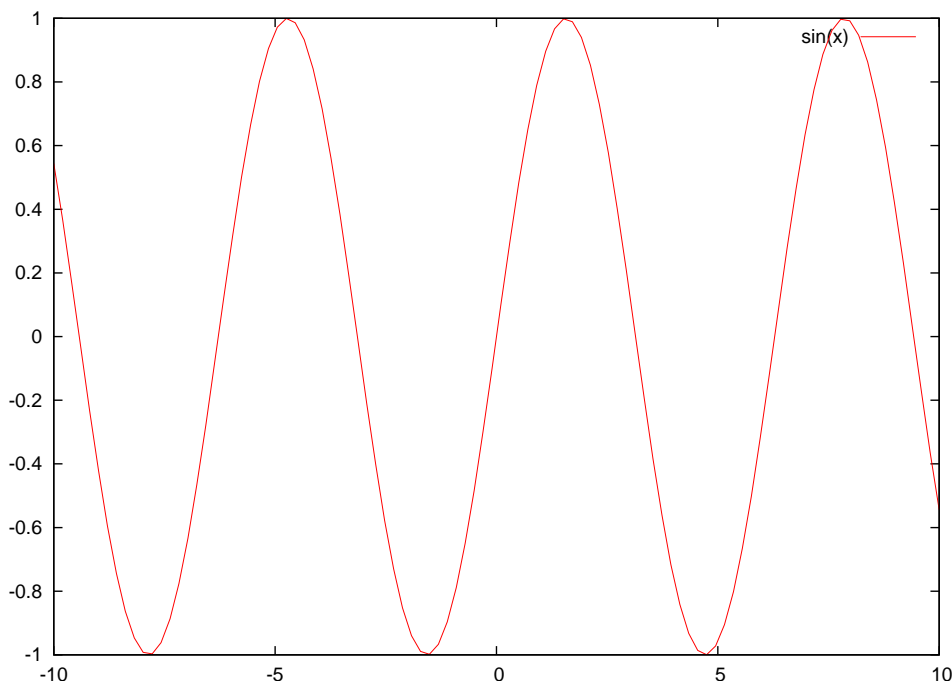
Začneme grafom funkcie $y = \sin x$. Na príkazovom riadku vidíme výzvu (tzv. prompt) v tvare

```
gnuplot> _
```

Sem napíšeme príkaz `plot sin(x)` (nezabudnite na zátvorky). Teda na obrazovke máme

```
gnuplot> plot sin(x)_
```

Po stlačení klávesu  sa na obrazovke objaví grafické okno. Vidíme ho na nasledujúcom obrázku¹



Obr. 1


¹Graf na obrazovke sa môže mierne odlišovať od tohto obrázku. Zdrojový text tejto brožúrky bol totiž vytvorený pomocou programu \TeX a následne konvertovaný do formátu PostScript. Z tohto dôvodu boli ilustračné obrázky exportované do formátu Encapsulated PostScript.

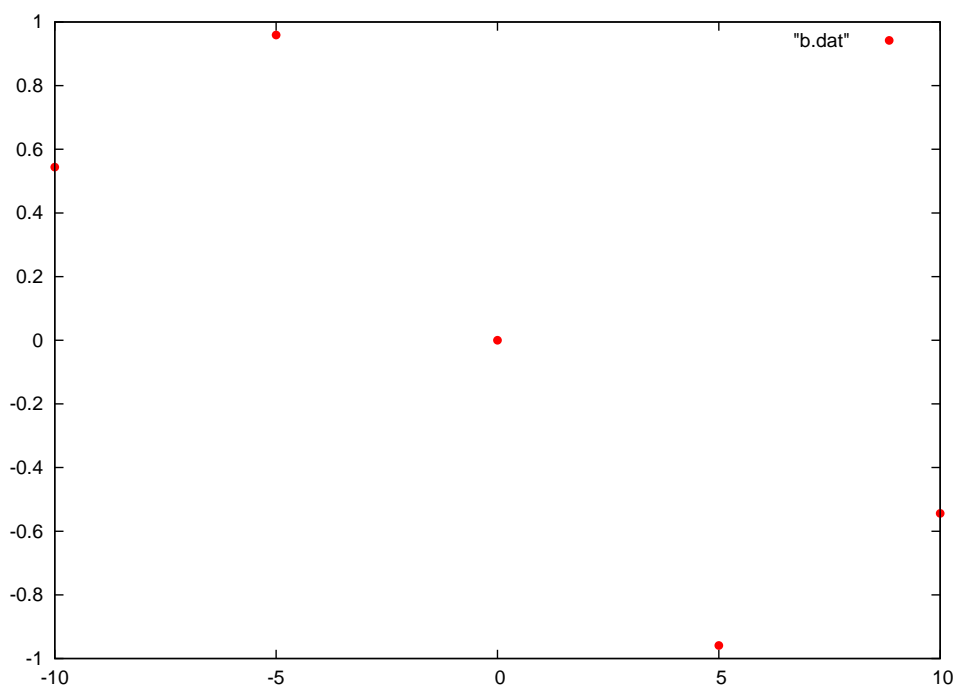
Teraz si nakreslíme graf funkcie danej tabuľkou. Budeme potrebovať dátový súbor, ktorý obsahuje dva stĺpce číselných údajov. Vytvoríme si napr. súbor s názvom `b.dat` (v nejakom ASCII editore², teda určite nie v MS Worde) s nasledujúcim obsahom

```
-10  0.544021
-5   0.958924
  0   0
  5  -0.958924
 10  -0.544021
```

Prvý stĺpec obsahuje hodnoty veličiny x , druhý stĺpec obsahuje im odpovedajúce hodnoty veličiny y . V programe `gnuplot` napíšeme príkaz `plot "b.dat"` (nezabudnite na úvodzovky). Teda na obrazovke máme

```
gnuplot> plot "b.dat" _
```

Po stlačení klávesu  sa na obrazovke objaví grafické okno. Vidíme ho na nasledujúcom obrázku



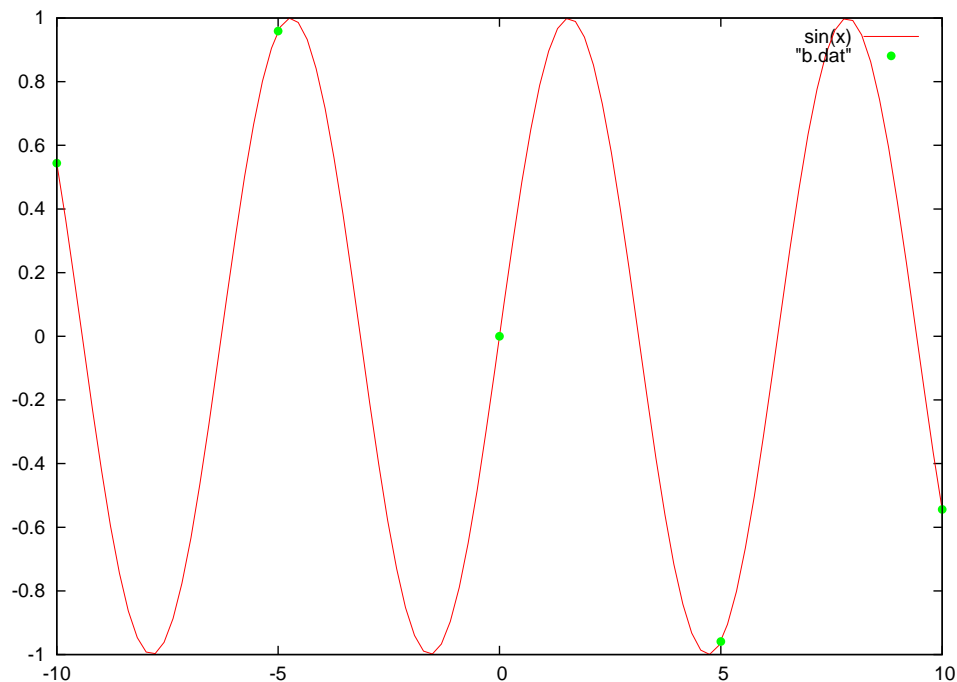
Obr. 2

Samozrejme, mohli sme obidva grafy nakresliť do jedného obrázku pomocou príkazu

```
gnuplot> plot sin(x), "b.dat"
```

Výsledok vidíme na nasledujúcom obrázku

²Môžete použiť napr. Poznámkový blok (Notepad).



Obr. 3

Cieľom tejto kapitoly bolo prvé oboznámenie sa s programom **gnuplot**. Aby to mohlo byť také jednoduché, autori programu definovali množstvo parametrov v rámci tzv. implicitných nastavení. Môžete sa o tom ľahko presvedčiť, keď si svoju prácu uložíte do súboru. Napríklad súbor **sinus.plt** vytvoríme pomocou príkazu `save "sinus.plt"`, teda takto

```
gnuplot> plot sin(x)
gnuplot> save "sinus.plt"
```

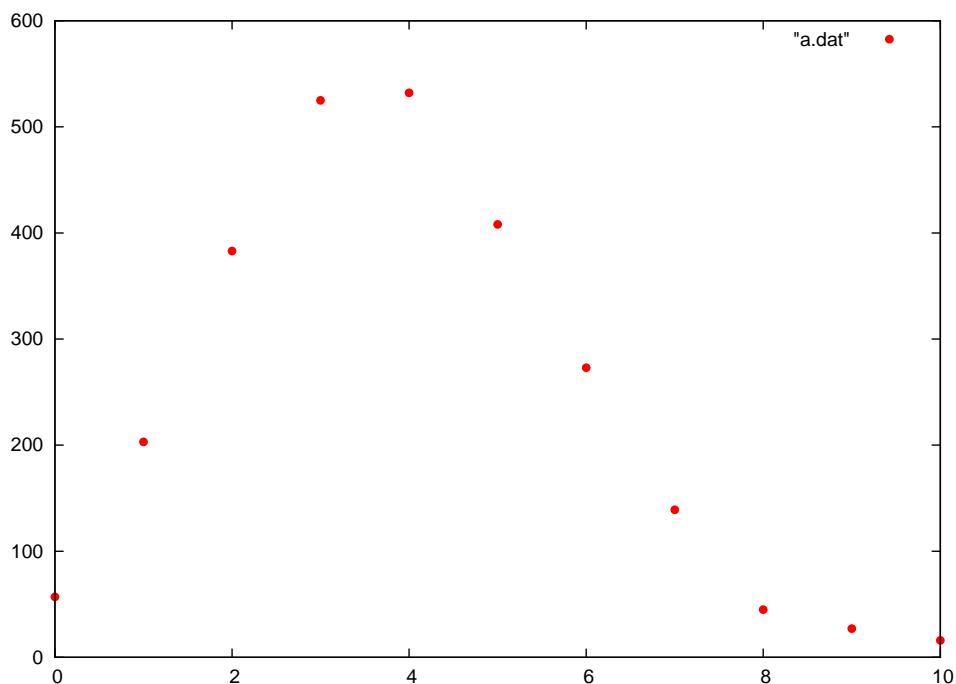
Teraz si už len stačí otvoriť súbor **sinus.plt** v nejakom ASCII editore. Uvidíte tam množstvo príkazov a nastavení, ktorým zatiaľ nemusíte rozumieť. Postupne sa niektoré z nich naučíme upravovať tak, aby sme výsledný graf prispôbili našim predstavám.

DÁTOVÉ SÚBORY

Takýto súbor nemusí obsahovať dva stĺpce, môže byť tvorený aj jediným stĺpcom. Ukážeme si to na príklade súboru `a.dat`, ktorý si vytvoríme v nejakom ASCII editore. Tento súbor obsahuje nasledujúci stĺpec číselných údajov³

```
# a.dat
57
203
383
525
532
408
273
139
45
27
16
```

Potom v programe `gnuplot` napíšeme príkaz `plot "a.dat"` (nezabudnite na úvodzovky).



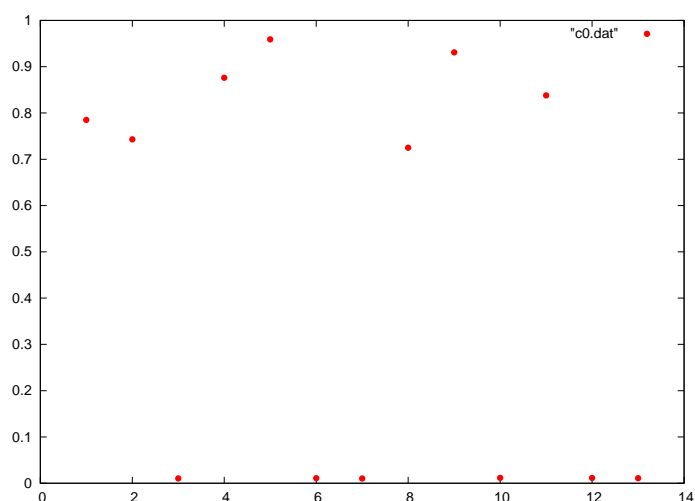
Obr. 4

`gnuplot` interpretoval tento stĺpec ako hodnoty veličiny na zvislej osi, pričom za hodnoty veličiny na vodorovnej osi vzal celé čísla od 0 po 10, t. j. presne tak isto, ako keby išlo o nasledujúci súbor

```
# a1.dat
0 57
1 203
2 383
3 525
4 532
5 408
6 273
7 139
8 45
9 27
10 16
```

Dátový súbor môže obsahovať aj viac stĺpcov. Napr. súbor `c0.dat` obsahuje nasledujúce číselné údaje

```
# c0.dat
1 7.85E-01 7 26 6 60
2 7.43E-01 1 29 15 52
3 1.043E-02 11 56 8 20
4 8.76E-01 11 31 8 47
5 9.59E-01 7 52 6 33
6 1.092E-02 11 55 9 22
7 1.027E-02 3 71 17 6
8 7.25E-01 1 31 22 44
9 9.31E-01 2 54 18 22
10 1.159E-02 21 47 4 26
11 8.38E-01 1 40 23 34
12 1.133E-02 11 66 9 12
13 1.094E-02 10 68 8 12
```



Obr. 5

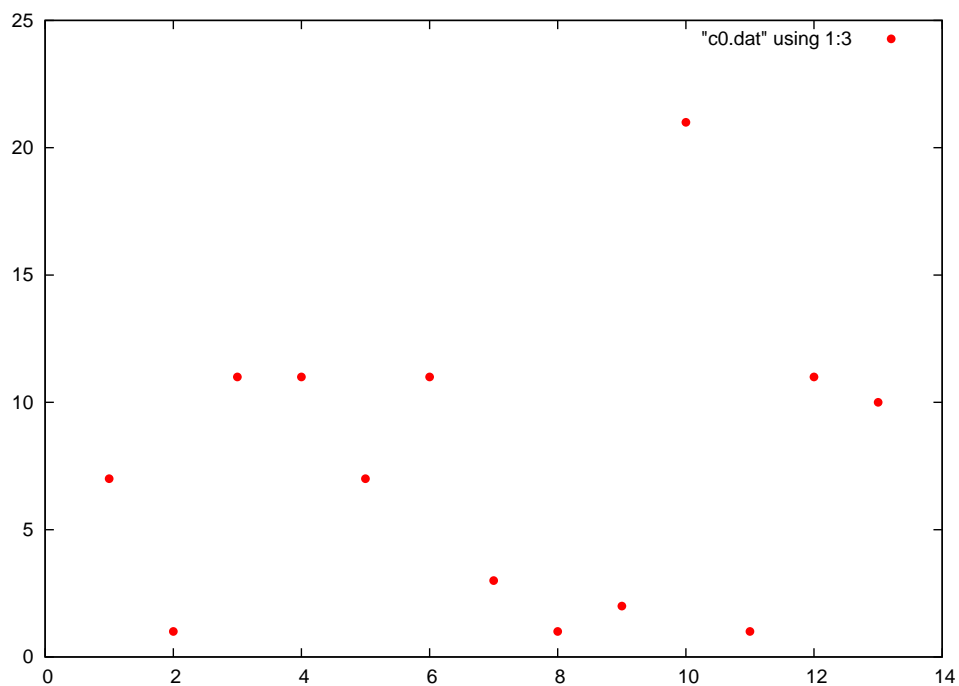
Ako vidíme na obrázku č. 5, `gnuplot` spracoval prvé dva stĺpce. Všimnime si, že číselné údaje v druhom

³Symbolom `#` začína komentár. Všetko od tohto symbolu až do konca riadku `gnuplot` ignoruje.

8

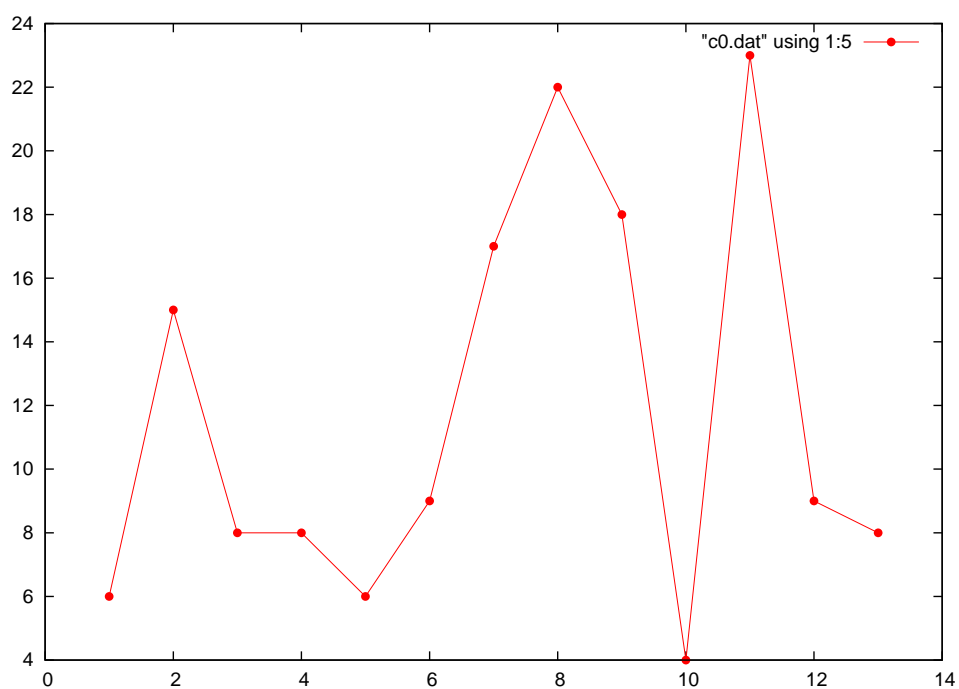
stĺpci sú zapísané v semilogaritmickej tvare. Napr. 7.85E-01 predstavuje číslo 7.85×10^{-1} , t. j. číslo 0.785.

Príkazom `plot "c0.dat" using 1:3` dosiahneme, že `gnuplot` vezme prvý a tretí stĺpec. Vidíme to na obrázku číslo 6.



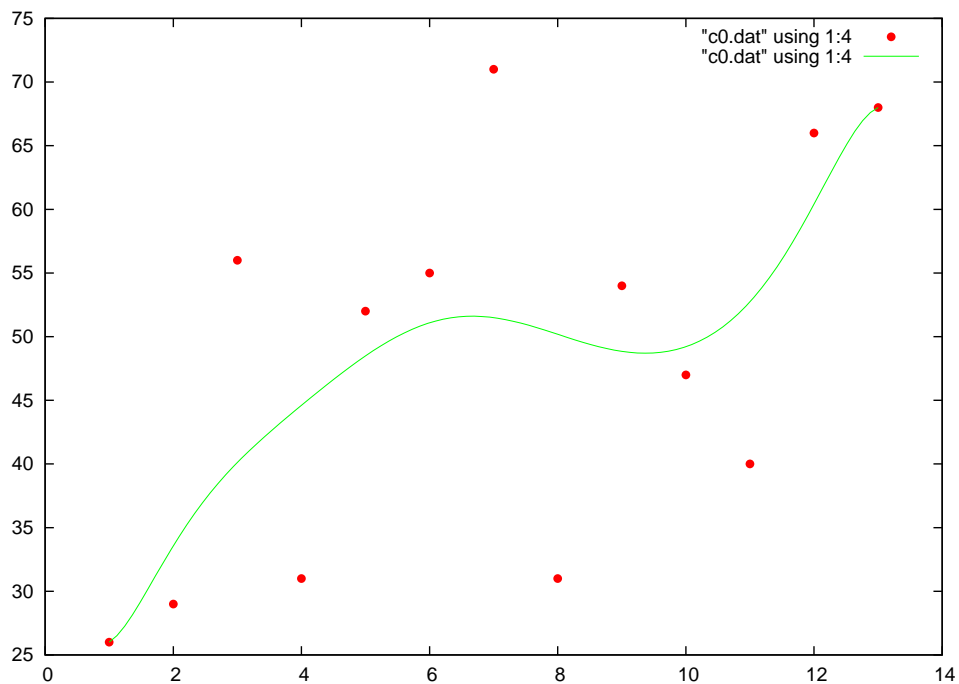
Obr. 6

Príkazom `plot "c0.dat" using 1:5 with linespoints` dosiahneme, že do bodového grafu `gnuplot` nakreslí aj polygón, t. j. spojí body grafu úsečkami. Vidíme to na obrázku číslo 7.



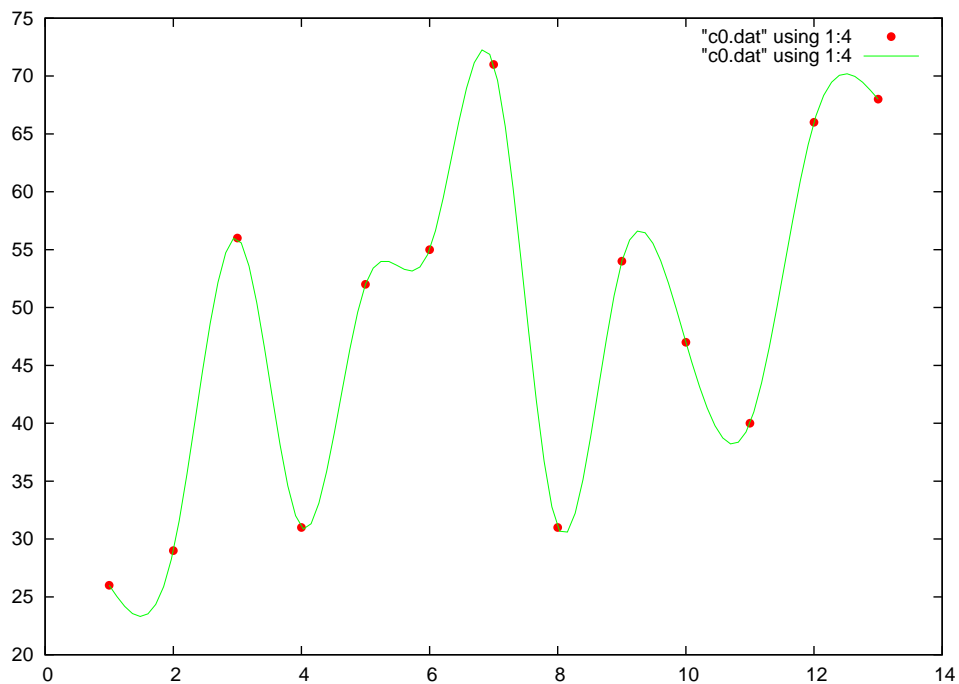
Obr. 7

Príkazom `plot "c0.dat" using 1:4 smooth bezier` dostaneme aproximáciu Bézierovou krivkou. Príkazom `plot "c0.dat" using 1:4, "c0.dat" using 1:4 smooth bezier` sme nakreslili obrázok číslo 8. Vidíme tam aj bodový graf, aj aproximáciu. Keďže sa v tomto príkaze názov súboru opakuje, stačí pri ďalších výskytoch tohto názvu písať namiesto "c0.dat" iba "". Teda v skutočnosti sme napísali príkaz v tvare `plot "c0.dat" using 1:4, "" using 1:4 smooth bezier`.



Obr. 8

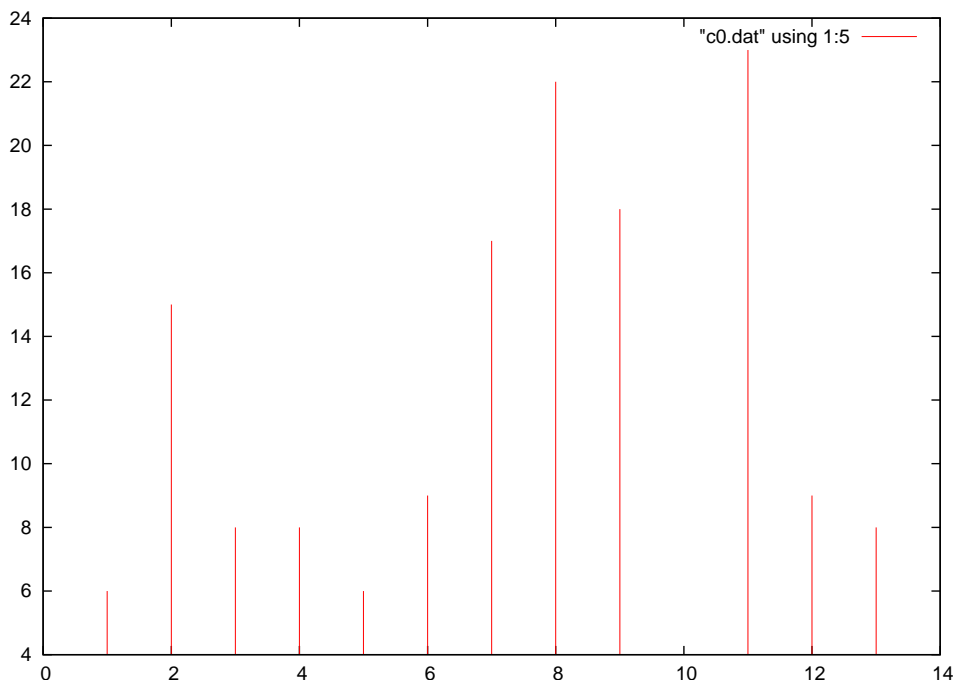
Podobne príkazom `plot "c0.dat" using 1:4, "" using 1:4 smooth csplines` dostaneme aproximáciu kubickým splajnom. Výsledok vidíme na obrázku číslo 9.



Obr. 9

10

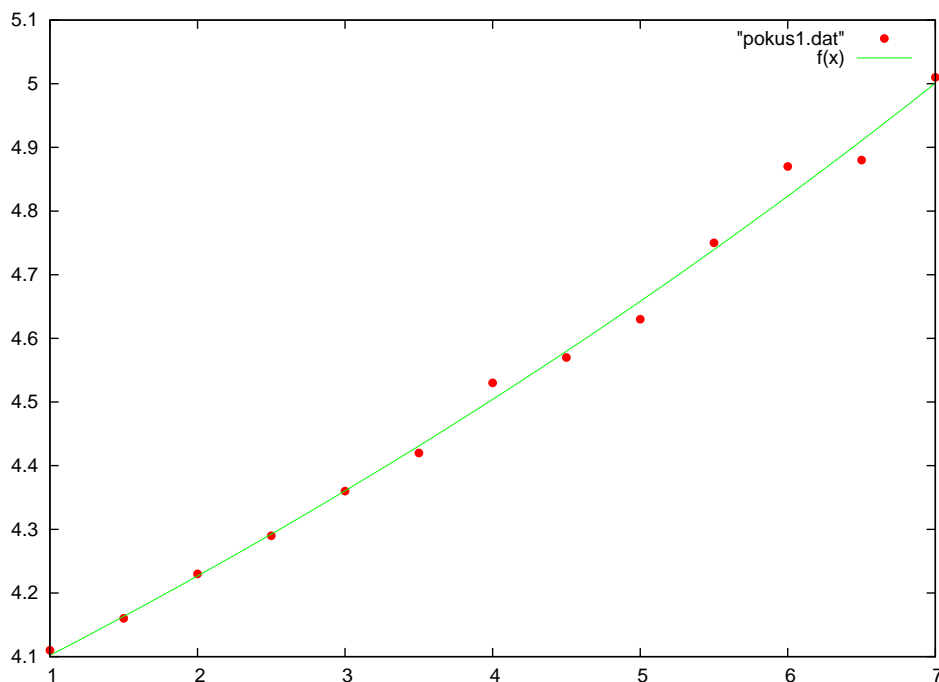
Stĺpcový diagram nakreslíme príkazom `plot "c0.dat" using 1:3 with impulses`. Výsledok je na obrázku číslo 10.



Obr. 10

Na obrázku číslo 11 vidíme vyrovnávanie nameraných údajov danou funkciou (tzv. fitovanie). Dátový súbor s názvom `pokus1.dat` obsahuje nasledujúce číselné údaje

```
# pokus1.dat
1.0 4.11
1.5 4.16
2.0 4.23
2.5 4.29
3.0 4.36
3.5 4.42
4.0 4.53
4.5 4.57
5.0 4.63
5.5 4.75
6.0 4.87
6.5 4.88
7.0 5.01
```



Obr. 11

Daná funkcia je určená predpisom $f(x) = a + b \cdot c^x$, pričom a , b , c sú neznáme konštanty. Na ich určenie sa používa nelineárna metóda najmenších štvorcov, konkrétne Marquardt-Levenbergov algoritmus. Ten vyžaduje štartovacie hodnoty pre neznáme konštanty, my použijeme $a=1$, $b=1$, $c=1$. V programe `gnuplot` napíšeme príkazy⁴

⁴Bodkočiarka slúži ako oddeľovač, umožňuje nám napísať viacero príkazov do jedného riadku.

```

a=1; b=1; c=1
f(x)=a+b*(c**x)
fit f(x) "pokus1.dat" via a, b, c
plot "pokus1.dat", f(x)

```

Môžeme používať aj multi-dátové súbory. Jednotlivé skupiny údajov sú oddelené dvojicou prázdnych riadkov. Napríklad súbor `d.dat` obsahuje nasledujúce skupiny číselných údajov

```

# d.dat
# index 0
532 2
537 6

# index 1
542 15

# index 2
547 22
552 26

# index 3
557 31
562 38
567 28
572 18
577 8
582 4
587 2

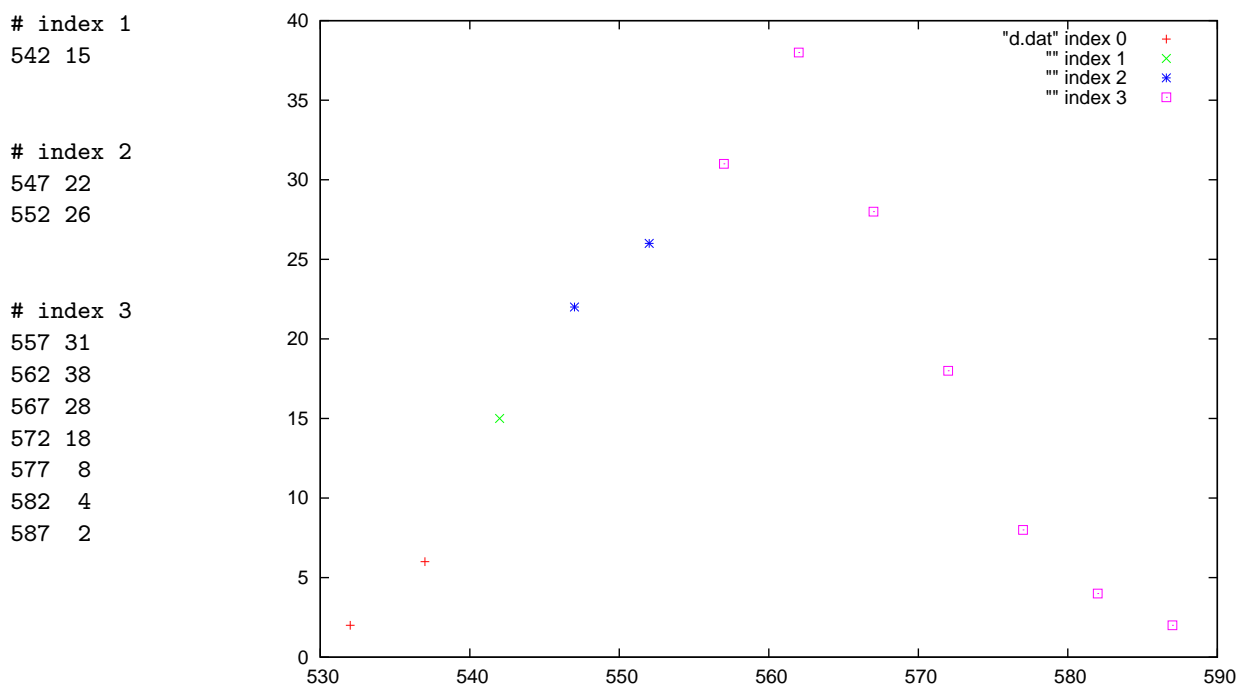
```

Pomocou nasledujúceho príkazu dosiahneme, že každá skupina je na grafe vykreslená inak.

```

plot "d.dat" index 0, "" index 1, "" index 2, "" index 3

```



Obr. 12

Všimnime si, že v súbore `c0.dat` sú jednotlivé číselné údaje v riadkoch oddelené medzerou.

```

# c1.dat
1, 7.85E-01, 7, 26, 6, 60
2, 7.43E-01, 1, 29, 15, 52
3, 1.043E-02, 11, 56, 8, 20
4, 8.76E-01, 11, 31, 8, 47
5, 9.59E-01, 7, 52, 6, 33
6, 1.092E-02, 11, 55, 9, 22
7, 1.027E-02, 3, 71, 17, 6
8, 7.25E-01, 1, 31, 22, 44
9, 9.31E-01, 2, 54, 18, 22
10, 1.159E-02, 21, 47, 4, 26
11, 8.38E-01, 1, 40, 23, 34
12, 1.133E-02, 11, 66, 9, 12
13, 1.094E-02, 10, 68, 8, 12

```

Pritom `gnuplot` si poradí aj s dátovými súbormi, v ktorých sa ako oddeľovač používa čiarka. Príkladom môže byť súbor `c1.dat`. Spracujeme ho príkazom `plot "c1.dat" using '%lf,%lf,%lf,%lf,%lf,%lf'`, kde sme `gnuplotu` poskytli informáciu o formátovaní riadkov. Výsledný graf bude identický s obrázkom číslo 5.

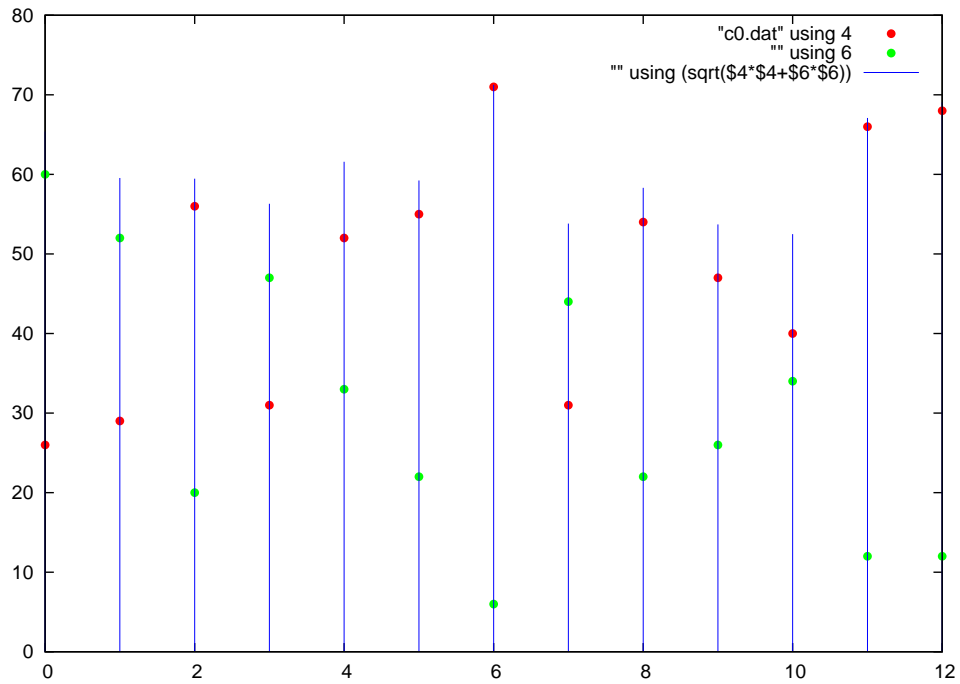
Jednotlivé stĺpce dátového súboru môžeme istým spôsobom kombinovať. Príkaz

```
plot "c0.dat" using 4, "" using 6, "" using (sqrt($4*$4+$6*$6)) with impulses
```

spôsobí, že na zvislej osi sa použijú hodnoty veličiny

$$y = \sqrt{x_4^2 + x_6^2}.$$

Pritom symbolom x_4 sme označili hodnoty zo štvrtého stĺpca a symbolom x_6 hodnoty zo šiesteho stĺpca súboru `c0.dat`.



Obr. 13

Grafy potrebujete pravdepodobne začleniť do nejakého dokumentu. V takomto prípade musíme zmeniť výstup programu `gnuplot`. K tomu nám slúži príkaz `set terminal`. Informácia o možnostiach výstupu je v `gnuplote` prístupná cez ponukovú lištu `File -> Show Output Devices`. V krátkosti sa zmienime iba o dvoch najpoužívanejších.

MS Word. V tejto časti si popíšeme cestu na začlenenie grafu do dokumentu formátu MS Word. Najskôr pomocou príkazu `set terminal cgm` presmerujeme výstup `gnuplotu` do formátu `cgm` (Computer Graphics Metafile). Ešte musíme určiť názov výstupného súboru príkazom `set output`. Ďalej používame `gnuplot` obvyklým spôsobom. Teda napríklad takto

```
set terminal cgm
set output "wdata.cgm"
set key 560, 12 spacing 3 box linetype 3 linewidth 4
plot "d.dat" index 0 title "data 0", "" index 1 title "data 1", \
"" index 2 title "data 2", "" index 3 title "data 3"
```

Ďalej budeme pracovať s vygenerovaným súborom `wdata.cgm`, prácu v programe `gnuplot` môžeme teda ukončiť. Otvoríme dokument formátu MS Word a vložíme do neho obrázok `wdata.cgm` cez ponukovú lištu `Vložiť -> Obrázok -> Zo súboru...` (`Insert -> Picture -> From File...`).

Niekedy je vhodnejší iný spôsob. V obrazovom okne `gnuplotu` klikneme pravým mišítkom na hornú lištu (má modrú farbu a je v ňom nápis `gnuplot graph`). Otvorí sa ponukové okno. Cez ponuku `Options -> Copy to Clipboard` uložíme obrázok do schránky Windows. Potom otvoríme dokument formátu MS Word a vložíme do neho obrázok cez klávesovú skratku `Ctrl+V`.

PostScript. Formát EPS (Encapsulated PostScript) sa používa pri začleňovaní obrázkov do dokumentov napísaných v `TeXu`. Najskôr pomocou príkazu `set terminal postscript` presmerujeme výstup `gnuplotu` do formátu PostScript. Ešte musíme určiť názov výstupného súboru príkazom `set output`. Ďalej používame `gnuplot` obvyklým spôsobom. Teda napríklad takto

```
set terminal postscript eps color solid
set output "po.eps"
set key 560, 12 spacing 3 box linetype 3 linewidth 4
plot "d.dat" index 0 title "data 0", "" index 1 title "data 1", \
"" index 2 title "data 2", "" index 3 title "data 3"
```

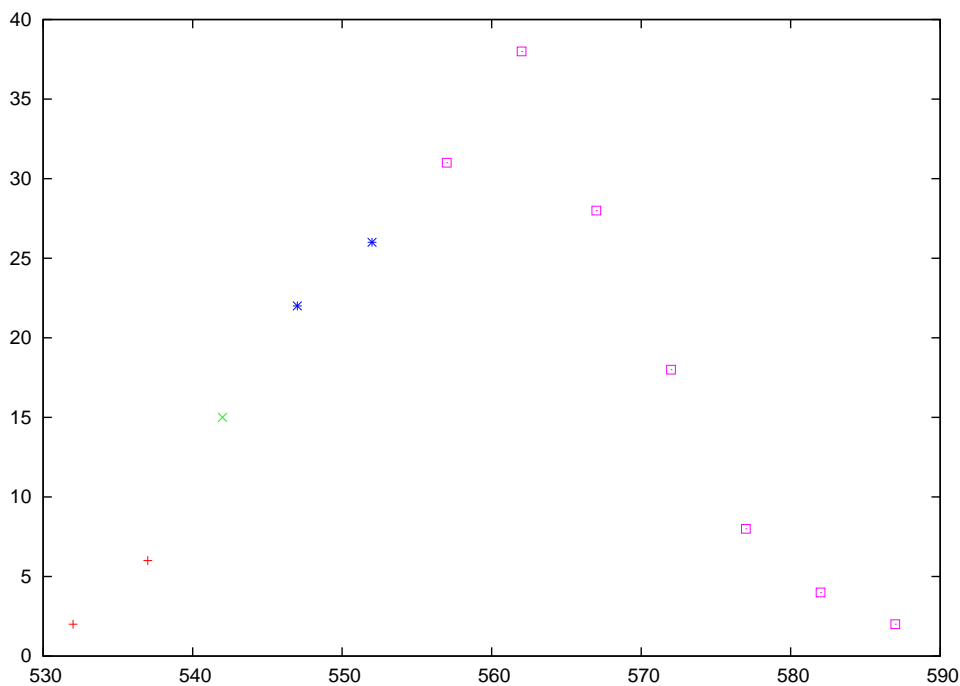
Ďalej budeme pracovať s vygenerovaným súborom `po.eps`, prácu v programe `gnuplot` môžeme teda ukončiť. Obrázok začleňujeme do dokumentu pomocou makra `epsf`.

Table. Príkazom `set terminal table` zabezpečíme, že `gnuplot` bude generovať dátový súbor. Tento výstup budeme používať pri kreslení grafov funkcií.

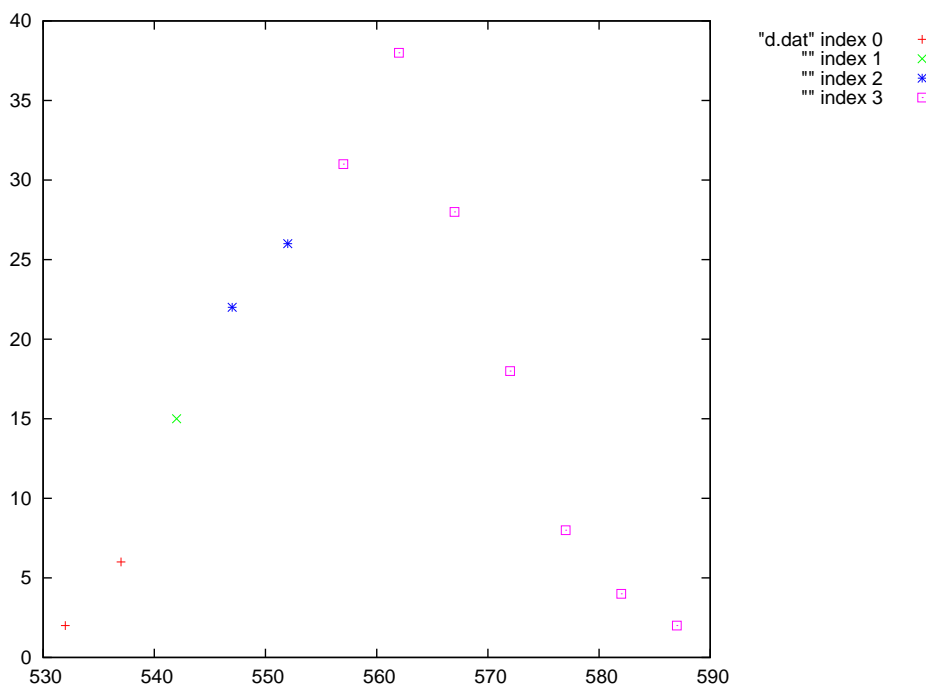
Pripomeňme si, že jednotlivé výstupy sa mierne odlišujú od toho, čo vidíme na obrazovke.

Legenda. Určíte ste si všimli, že naše obrázky obsahovali stručný popis zobrazovaných údajov, tzv. legendu. Autori **gnuplotu** ju umiestnili (v rámci implicitných nastavení) do pravého horného rohu. Tu však môže prekážať. Vtedy ju presunieme na iné miesto, prípadne odstránime úplne.

Zobrazovanie legendy môžeme zakázať príkazom **unset key** (obrázok číslo 14), prípadne umiestniť legendu mimo plochy grafu príkazom **set key outside** (obrázok číslo 15).



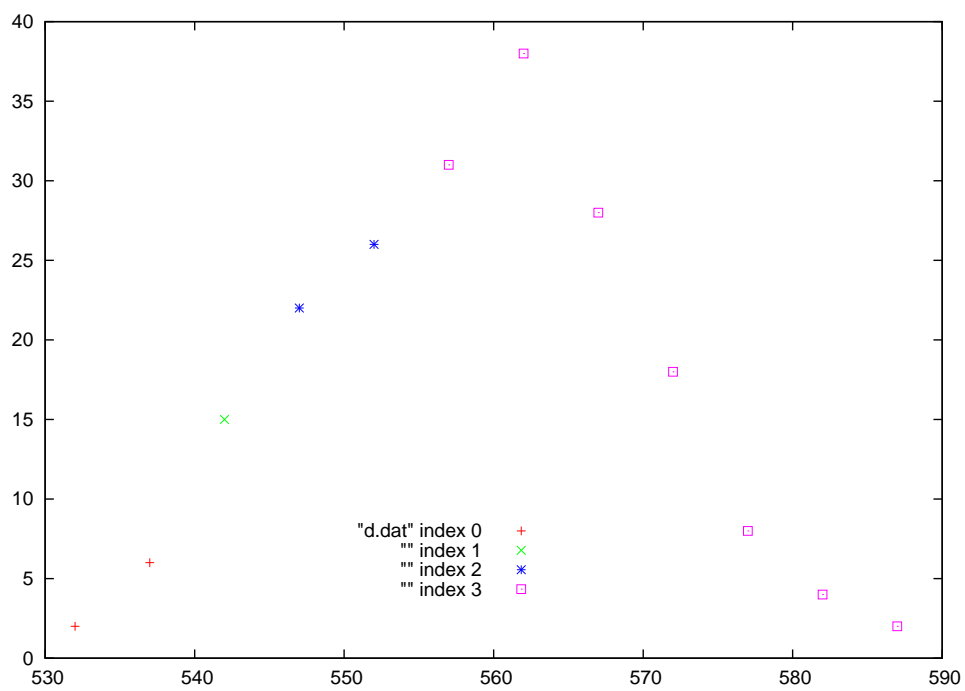
Obr. 14



Obr. 15

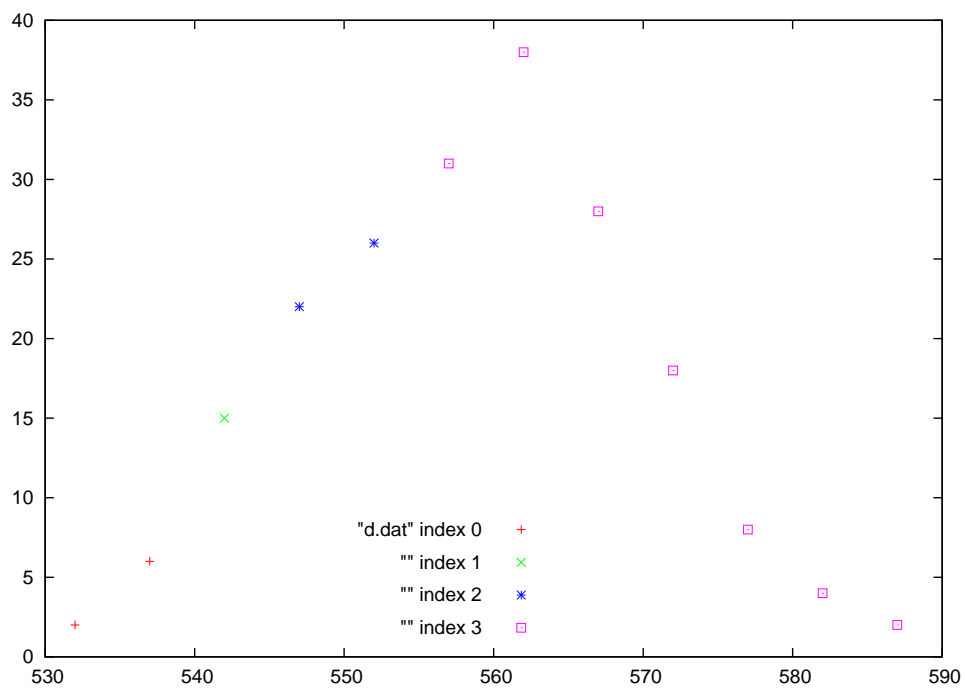
Ako vidíme na obrázku číslo 15, **gnuplot** kvôli legende zmenil obdĺžnik určený pre samotný graf. Pre celý obrázok má totiž vyčlenený obdĺžnik, ktorého strany sú v pomere 5 : 3.

Príkazom `set key 560, 8` presunieme legendu do daného bodu (obrázok číslo 16).



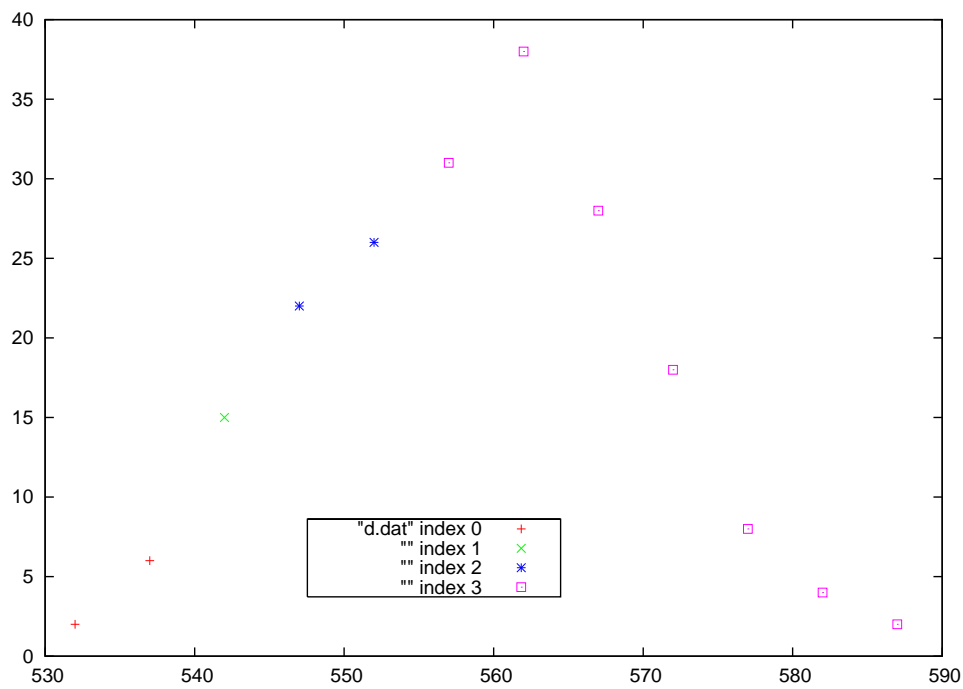
Obr. 16

Príkazom `set key 560, 8 spacing 3` navyše zväčšíme v legende vzdialenosť medzi riadkami (obrázok číslo 17).



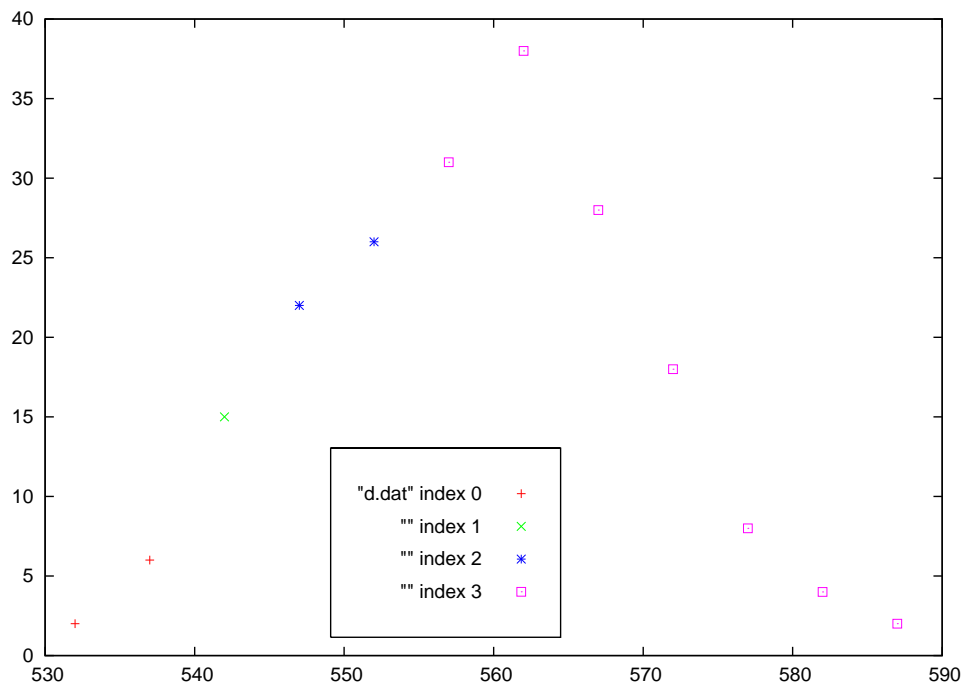
Obr. 17

Príkazom `set key 560, 8 box` nakreslíme legendu v rámečku (obrázok číslo 18).



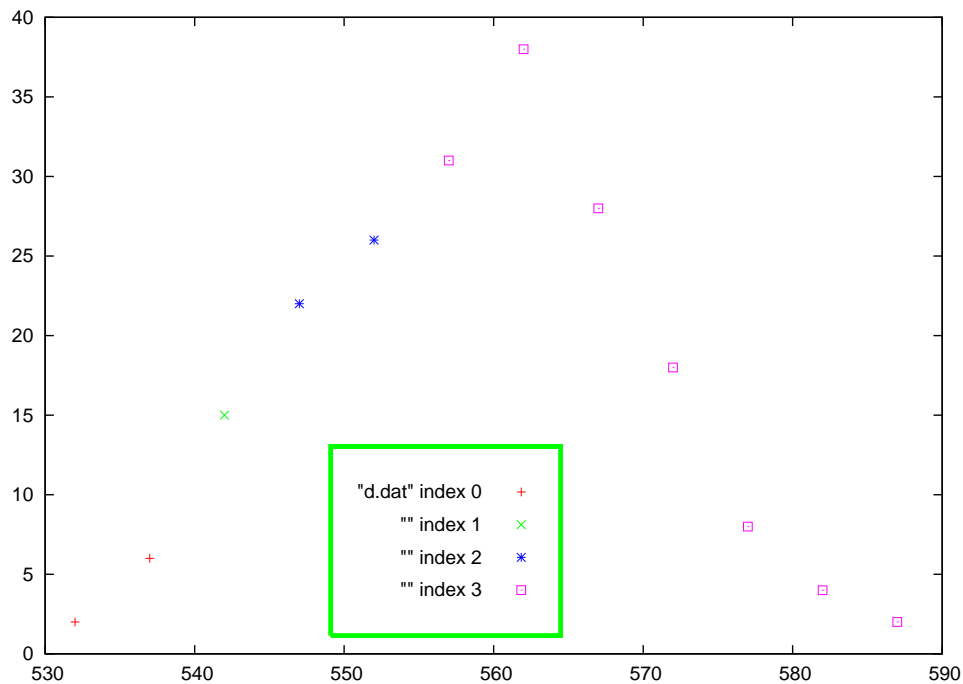
Obr. 18

Príkazom `set key 560, 12 spacing 3 width -2 height 3 box` tento rámeček trochu vylepšíme (obrázok číslo 19).



Obr. 19

Pomocou príkazu `set key 560, 12 spacing 3 width -2 height 3 box linestyle 2 linewidth 7` zmeníme typ a hrúbku čiar rámčeka (obrázok číslo 20).

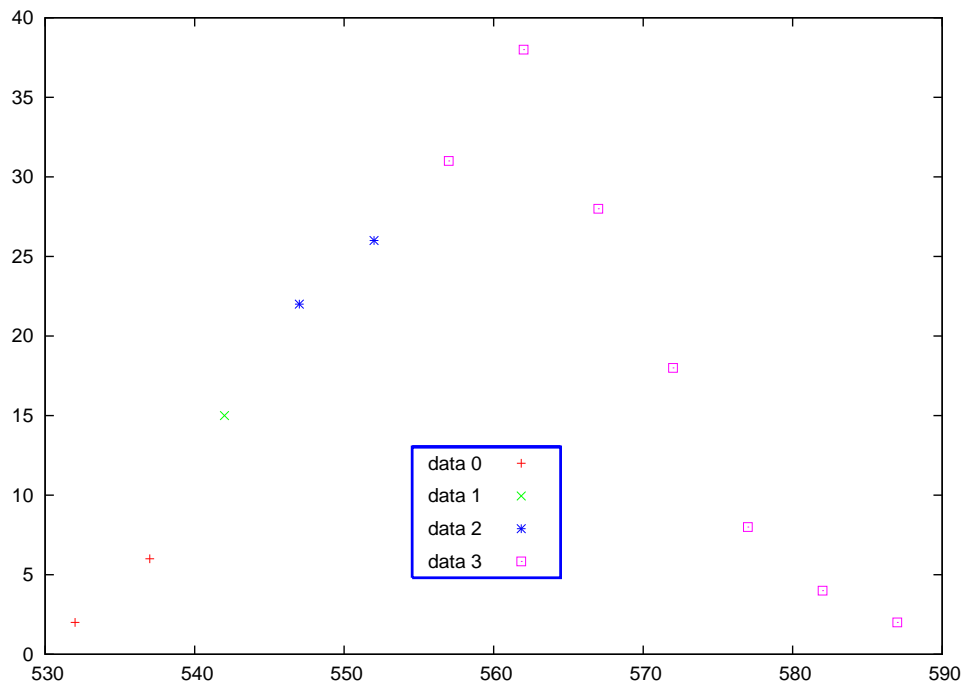


Obr. 20

Môžeme zmeniť aj text legendy pomocou príkazu `title`, ktorý použijeme v rámci príkazu `plot`. Tu je ukážka

```
set key 560, 12 spacing 3 box linestyle 3 linewidth 4
plot "d.dat" index 0 title "data 0", "" index 1 title "data 1", \
"" index 2 title "data 2", "" index 3 title "data 3"
```

Obrátené lomítko (backslash) slúži na rozdelenie dlhšieho príkazu do viacerých riadkov. Výsledok je na obrázku číslo 21.



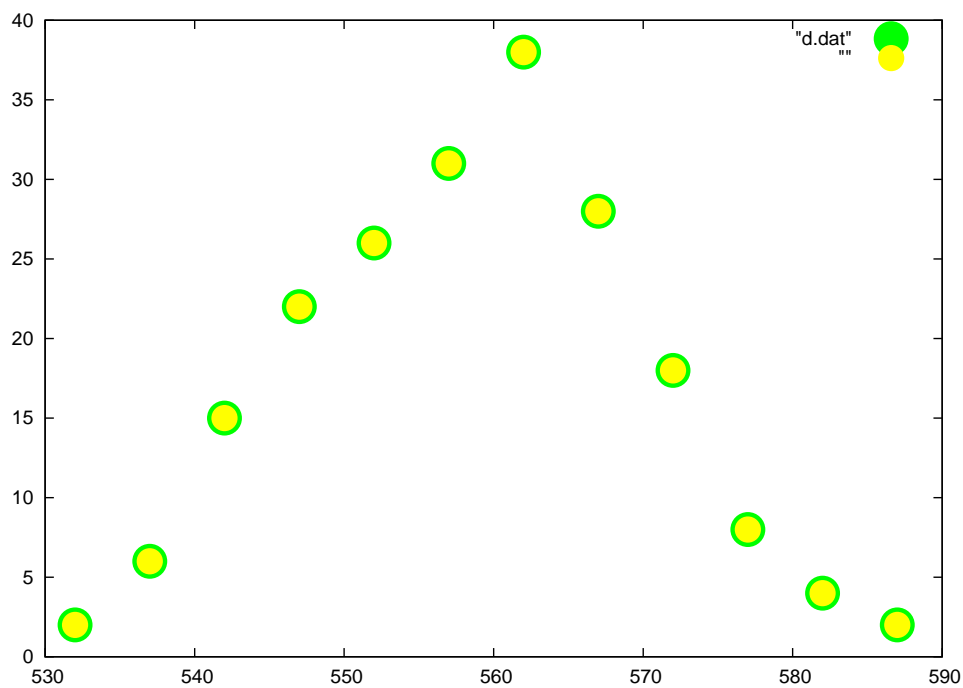
Obr. 21

Body a čiary.

V bodovom grafe môžeme nastaviť typ symbolov a ich veľkosť⁵. Napríklad pomocou príkazu

```
plot "d.dat" with points linetype 2 pointtype 7 pointsize 4,\
"" with points linetype 6 pointtype 7 pointsize 3
```

dostaneme graf, ktorý vidíte na obrázku číslo 22.

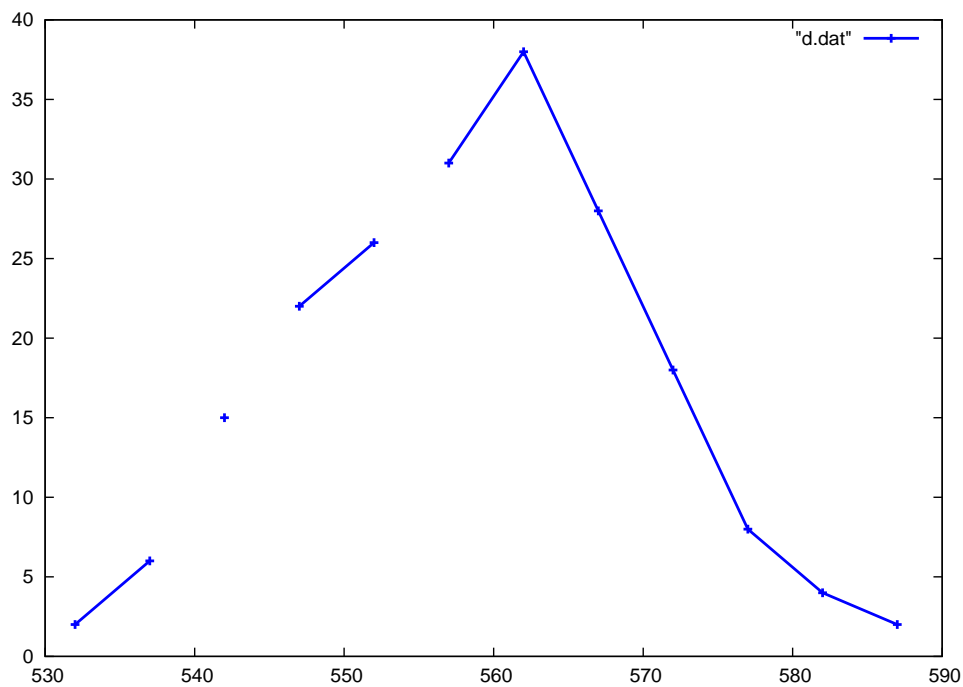


Obr. 22

V spojnicovom grafe môžeme nastaviť typ čiary a jej hrúbku. Napríklad pomocou príkazu

```
plot "d.dat" with linespoints linetype 3 linewidth 4
```

dostaneme graf, ktorý vidíte na obrázku číslo 23.



Obr. 23

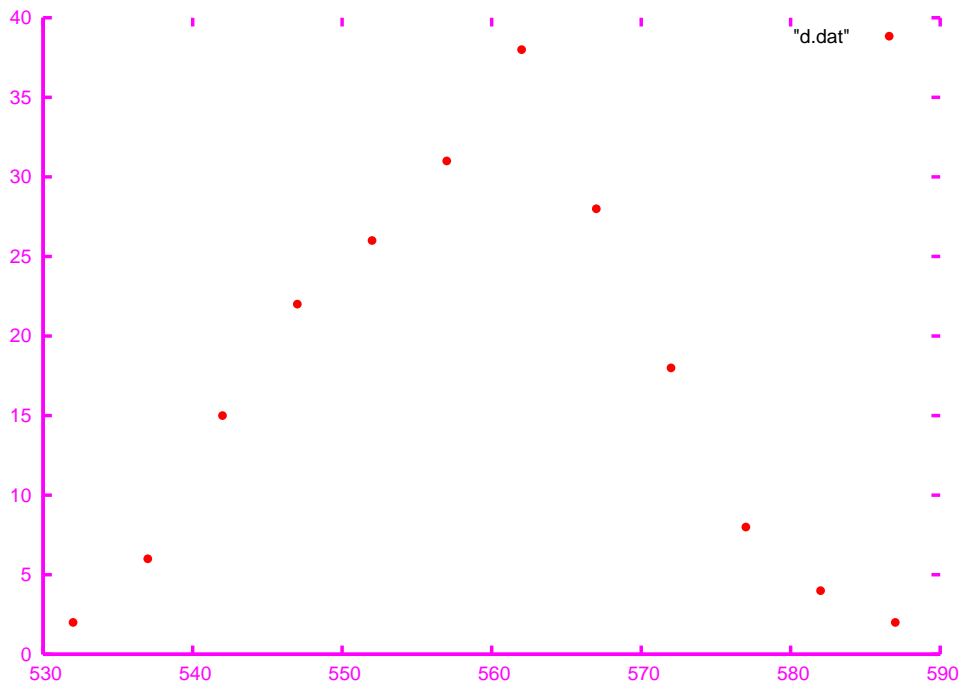
⁵Informáciu o jednotlivých možnostiach sa v **gnuplot**e dozvieme príkazom **test**.

Rámček. Medzi implicitné nastavenia patrí aj rámček okolo grafu. Môžeme ho úplne odstrániť príkazom `unset border`, prípadne modifikovať podľa našich potrieb.

Jednotlivé strany rámčeka sú očíslované mocninami dvojky: dolná 1, ľavá 2, horná 4, pravá 8. Ak chceme vykresliť iba niektoré z nich, v príkaze `set border` uvidíme číslo, ktoré dostaneme ako súčet čísel priradených týmto stranám.

Napríklad dolná a ľavá strana majú súčet čísel $1 + 2 = 3$. Pomocou príkazu `set border 3 linestyle 4 linewidth 5`

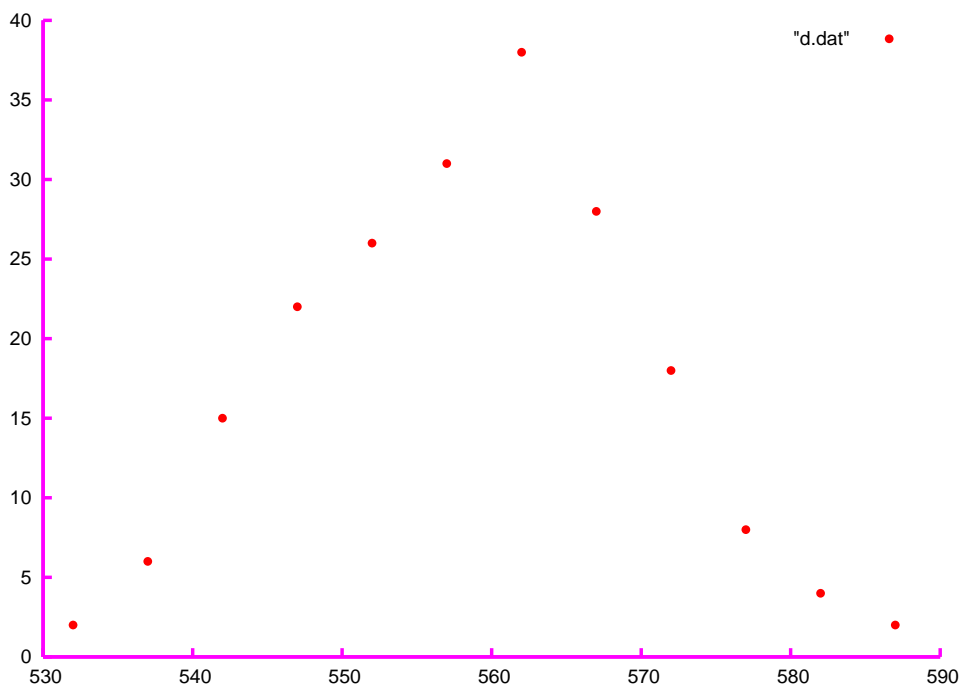
dostaneme obrázok číslo 24.



Obr. 24

Hore a vpravo zostali dielky stupníc, sú akosi navyše. Napravíme to príkazmi `set xtics nomirror textcolor lt -1`; `set ytics nomirror textcolor lt -1`

Výsledok vidíme na obrázku číslo 25.

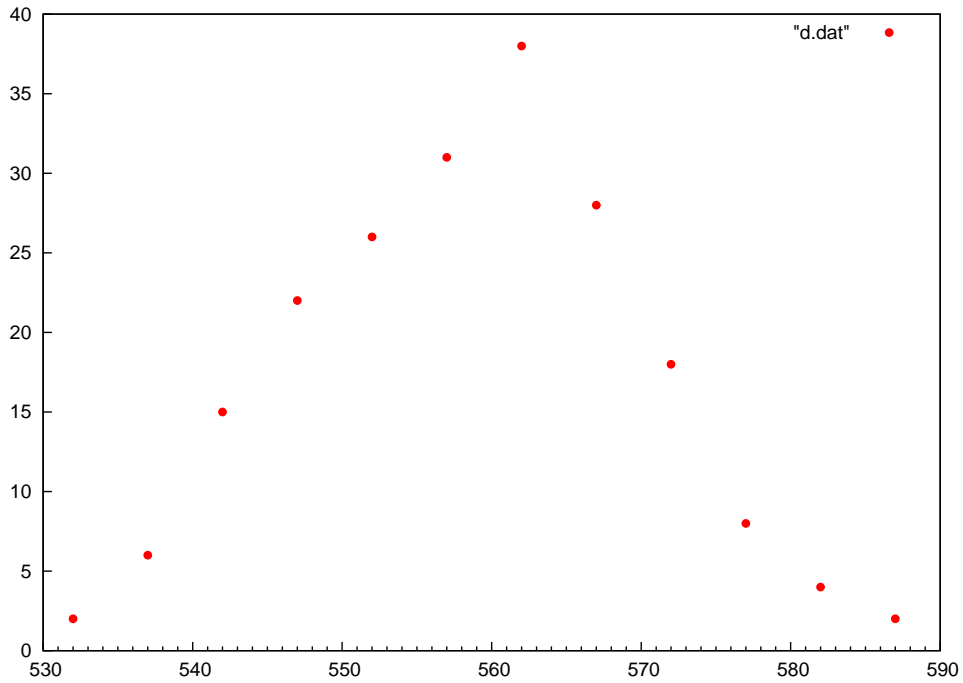


Obr. 25

Stupnice môžeme zahustiť menšími dielikmi pomocou príkazov `set mxtics`, `set mytics`. Ukážeme si to na príklade

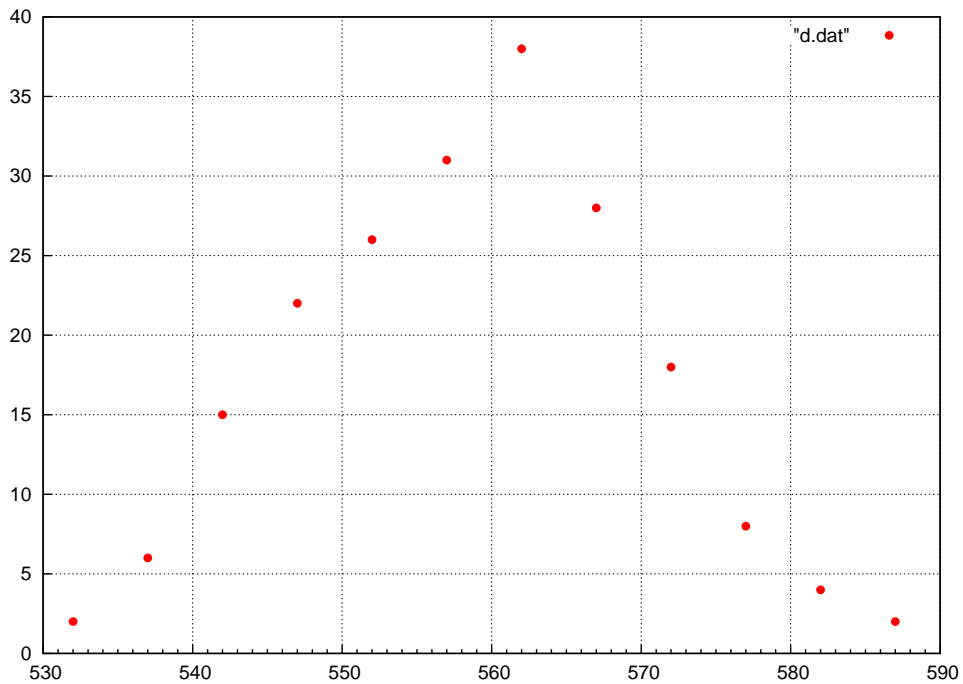
```
set xtics nomirror
set ytics nomirror
set mxtics 10
plot "d.dat"
```

Výsledok vidíme na obrázku číslo 26.



Obr. 26

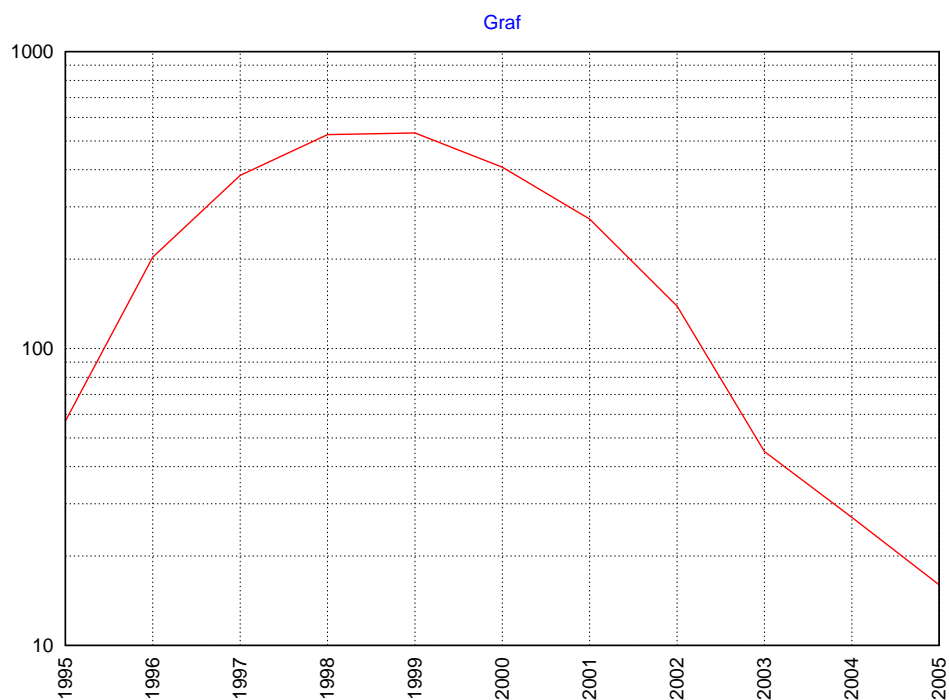
Mriežku pridáme príkazom `set grid`. Výsledok vidíme na obrázku číslo 27.



Obr. 27

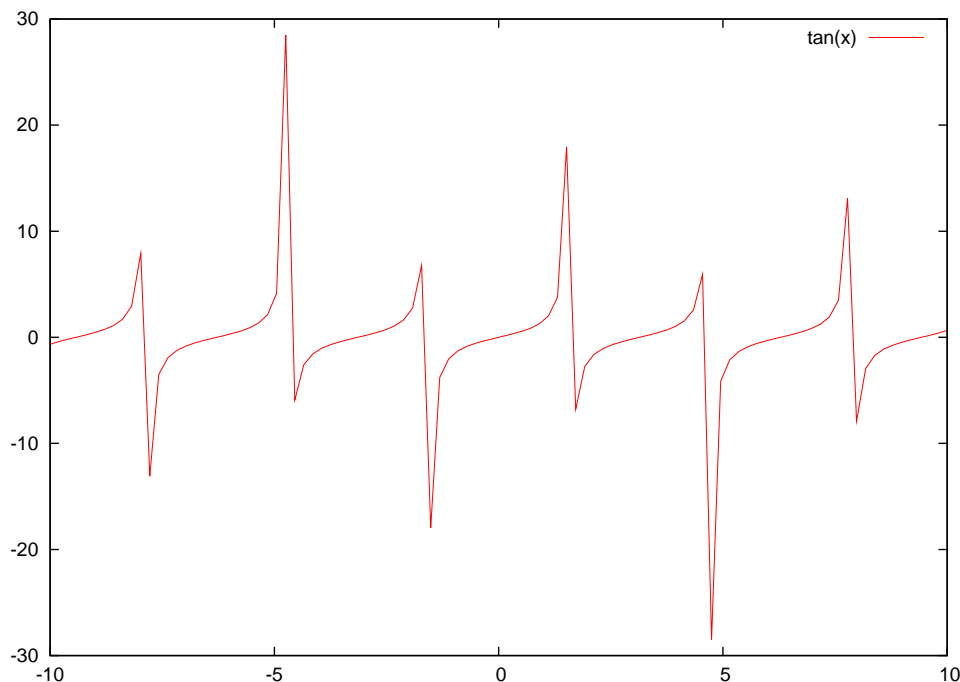
Logaritmicкую stupnicu na zvislej osi, pomenovanie dielikov na vodorovnej osi, ako aj niektoré ďalšie drobnosti ilustruje nasledujúci príklad. Výsledok vidíme na obrázku číslo 28.

```
set xtics rotate ('1995'0,'1996'1,'1997'2,'1998'3,\
'1999'4,'2000'5,'2001'6,'2002'7,'2003'8,'2004'9,'2005'10)
set ytics 10
unset mxtics
set mytics 10
unset grid
set logscale y 10
set title 'Graf' tc lt 3
set samples 1000
unset key
set ticscale 0
set grid xtics mytics
plot [0:10][10:1000] "a.dat" with lines linetype 1 linewidth 2
```



Obr. 28

V úvodnej časti sme nakreslili graf funkcie sínus, teraz sa pokúsime nakresliť graf funkcie tangens. Už kalkulačky nás naučili, že desatinná čiarka je vlastne bodka. Tiež tangens nie je tg, ale tan. Použijeme teda príkaz `plot tan(x)` (nezabudnite na zátvorky). Výsledok vidíme na obrázku číslo 29.



Obr. 29

Pokúsime sa tomuto obrázku porozumieť⁶. Pomocou príkazov

```
set terminal table
set output "tangens.dat"
plot tan(x)
```

`gnuplot` vyrobí dátový súbor `tangens.dat`, ktorý si môžete prezrieť nejakým ASCII editorom. Prvý stĺpec obsahuje 100 čísel⁷ rovnomerne rozložených od -10 do 10. Im odpovedajúce hodnoty funkcie tangens sú v druhom stĺpci. Obrázok číslo 29 môžeme teraz nakresliť aj pomocou príkazu

```
plot "tangens.dat" with lines
```

Pritom `gnuplot` nastaví implicitne vodorovnú os v rozsahu od -10 do 10 a zvislú os prispôsobí vypočítaným hodnotám (vidíme ich v súbore `tangens.dat`).

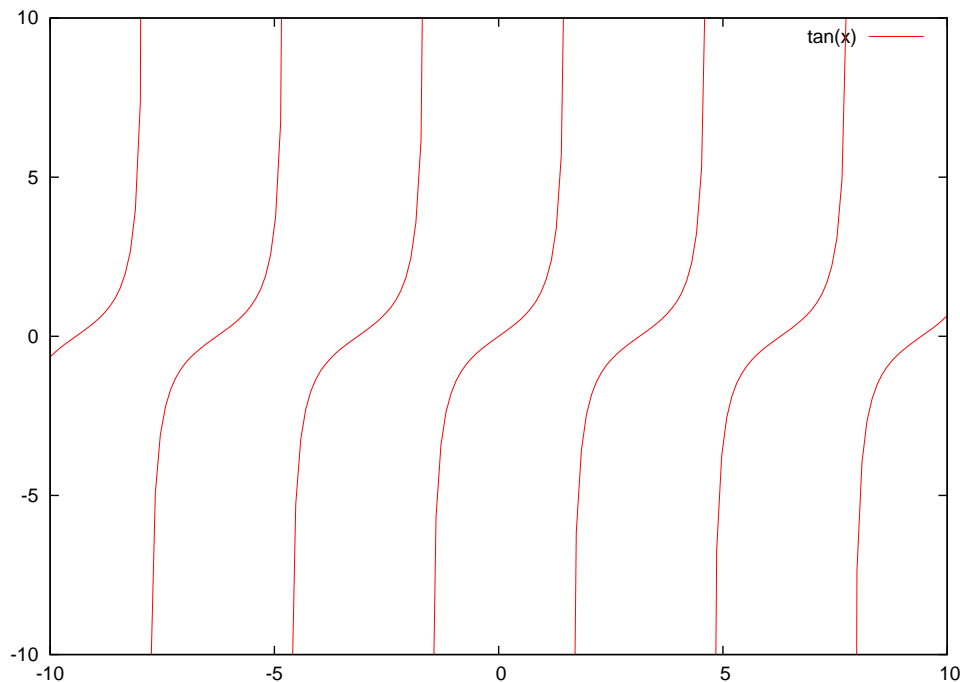
Teraz už môžeme zmenou nastavení dosiahnuť prijateľnejší obrázok. Použijeme príkazy

```
set yrange [-10:10]
set sample 180
plot tan(x)
```

Výsledok vidíme na obrázku číslo 30.

⁶V elektronickej verzii tejto brožúrky sa môžeme presvedčiť o tom, že graf na obrázku číslo 29 je vlastne lomená čiara zložená z úsečiek (už pri zväšení na 500%).

⁷Tieto čísla majú tvar $x_i = -10 + ih$, kde h je krok a $i = 0, 1, 2, \dots, 99$. Pretože posledné z nich je číslo 10, platí $x_{99} = -10 + 99h = 10$. Odtiaľ vypočítame $h = \frac{20}{99} \doteq 0.20202$.



Obr. 30

Znovu vyrobíme dátový súbor pomocou príkazov

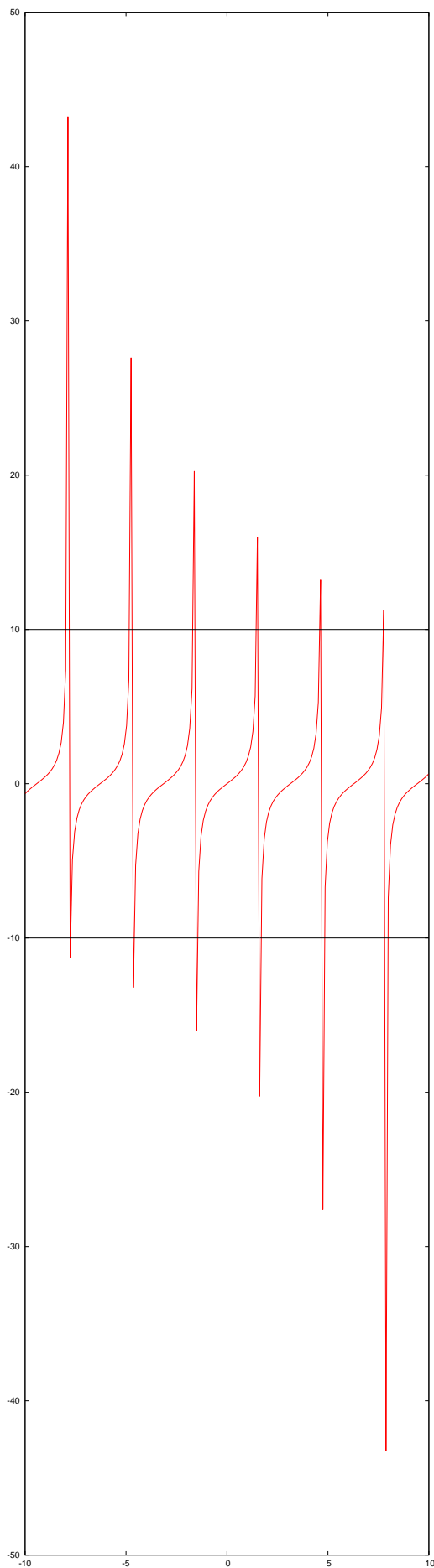
```
set yrange [-10:10]
set sample 180
set terminal table
set output "tangens.dat"
plot tan(x)
```

V dátovom súbore `tangens.dat` si teraz pozrieme tretí stĺpec. Obsahuje buď symbol `i`, ak sa daný bod (určený dvojicou čísel v tom istom riadku) nachádza v obrazovom poli, alebo obsahuje symbol `o`, ak daný bod leží mimo obrazového poľa.

Ak `gnuplot` spája úsečkou dva body, z ktorých jeden leží v obrazovom poli a druhý mimo tohto obrazového poľa, vykreslí sa iba časť úsečky ležiaca v tomto obrazovom poli. Názorne si to môžeme predstaviť ako urobenie výrezu (pozri obrázok číslo 31). Jediný rozdiel oproti obrázku číslo 30 je v tom, že `gnuplot` nespojí dva body, ak obidva ležia mimo obrazového poľa.

Pre zaujímavosť uvedieme príkazy, pomocou ktorých sme nakreslili obrázok číslo 31.

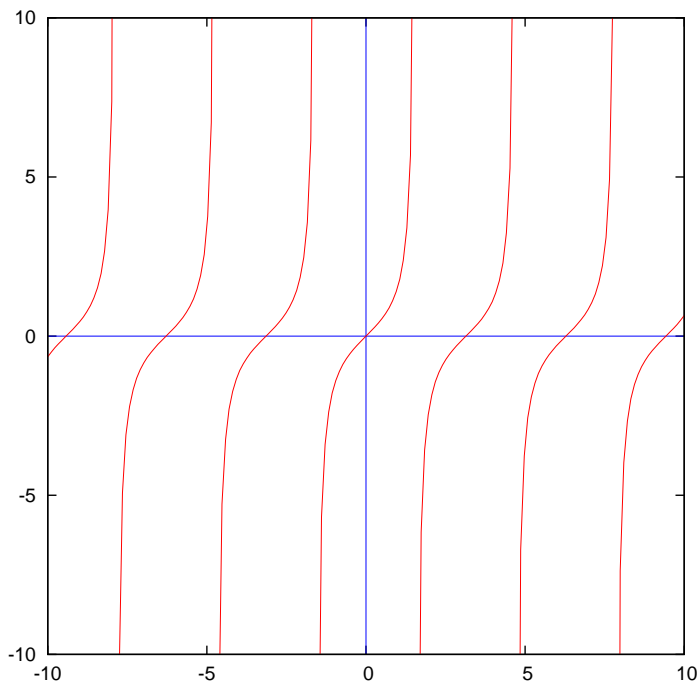
```
set sample 180
set size 1,5
unset key
plot tan(x), -10 linetype -1, 10 linetype -1
```



Obr. 31

Osi nakreslíme pomocou príkazu `set zeroaxis`. Rámček v tvare štvorca dosiahneme príkazom `set size square`. Obrázok číslo 32 bol nakreslený pomocou príkazov

```
set sample 180
unset key
set yrange [-10:10]
set zeroaxis linetype 3
set size square
plot tan(x)
```



Obr. 32

Rámček a stupnice odstránime príkazmi

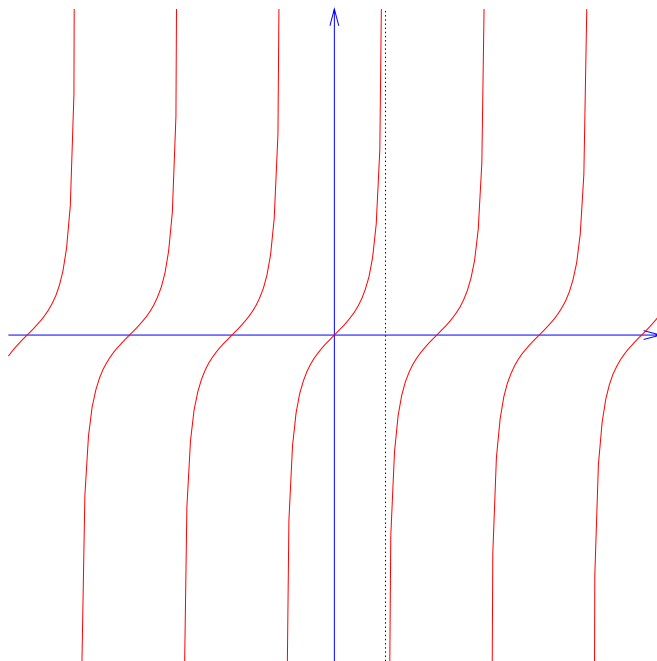
```
unset border; unset xtics; unset ytics
```

Osi môžeme narysovať aj ako orientované úsečky

```
set arrow 1 from -10,0 to 10,0
set arrow 2 from 0,-10 to 0,10
```

K dokonalosti tomu ešte chýbajú asymptoty, nakreslíme si na ukážku aspoň jednu. Obrázok číslo 33 bol vytvorený pomocou príkazov

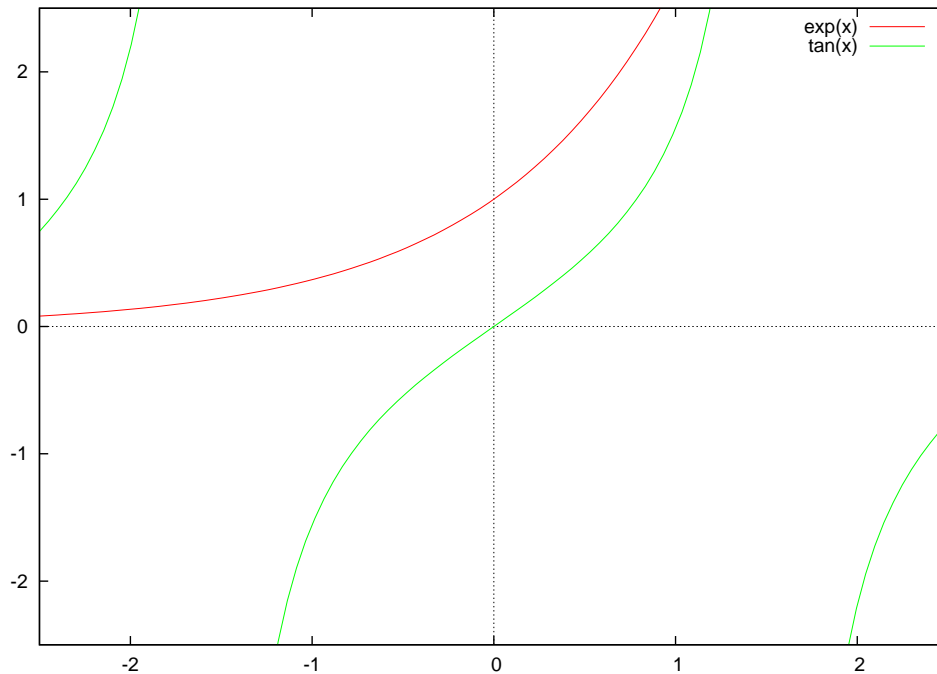
```
unset key; unset border
unset xtics; unset ytics
set sample 180
set yrange [-10:10]
set arrow 1 from -10,0 to 10,0 linetype 3
set arrow 2 from 0,-10 to 0,10 linetype 3
set arrow 3 from pi/2,-10 to pi/2,10 \
nohead linetype 0
set size square
plot tan(x)
```

Obr. 33

Do jedného obrázku môžeme nakresliť aj niekoľko grafov funkcií. Na obrázku číslo 34 vidíme graf exponenciálnej funkcie, ako aj graf funkcie tangens. Použili sme príkazy

```
set xrange [-2.5:2.5]
set yrange [-2.5:2.5]
set zeroaxis
plot exp(x), tan(x)
```



Obr. 34

Ďalšia ukážka sa bude týkať funkcie

$$f(x) = \begin{cases} x^2 & -1 \leq x < 1 \\ 2 - x & 1 \leq x \end{cases}$$

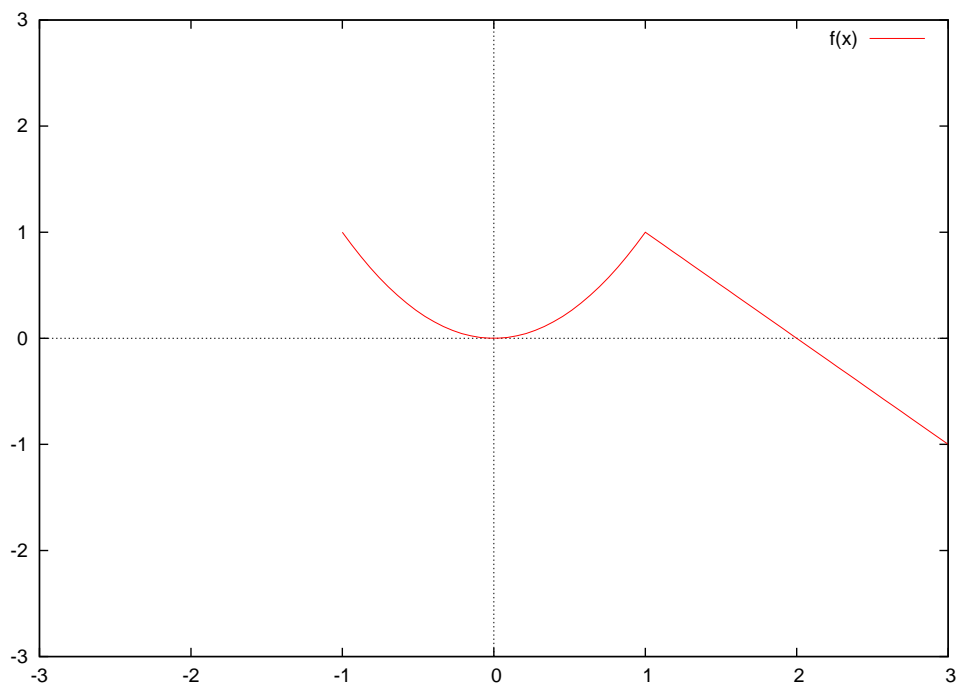
ktorej graf nakreslíme pomocou príkazov

```
set zeroaxis
f(x) = -1<=x && x<1 ? x**2 : 1<=x ? 2-x : 1/0
set xrange [-3:3]
set yrange [-3:3]
plot f(x)
```

Použili sme ternárny operátor $a?b:c$, ktorý funguje podobne ako v jazyku C. Ak a je pravdivé, vykoná sa b , v opačnom prípade sa vykoná c .

Výraz $1/0$ používame tam, kde funkcia nie je definovaná.

Výsledok vidíme na obrázku číslo 35.



Obr. 35

Ten istý výsledok dosiahneme aj príkazmi

```
set zeroaxis
set xrange [-3:3]
set yrange [-3:3]
g(x) = x>=-1 ? x**2 : 1/0
f(x) = x<1 ? g(x) : 2-x
plot f(x)
```

Nespojité funkcie je vhodné rozdeliť na niekoľko častí. Napríklad graf funkcie

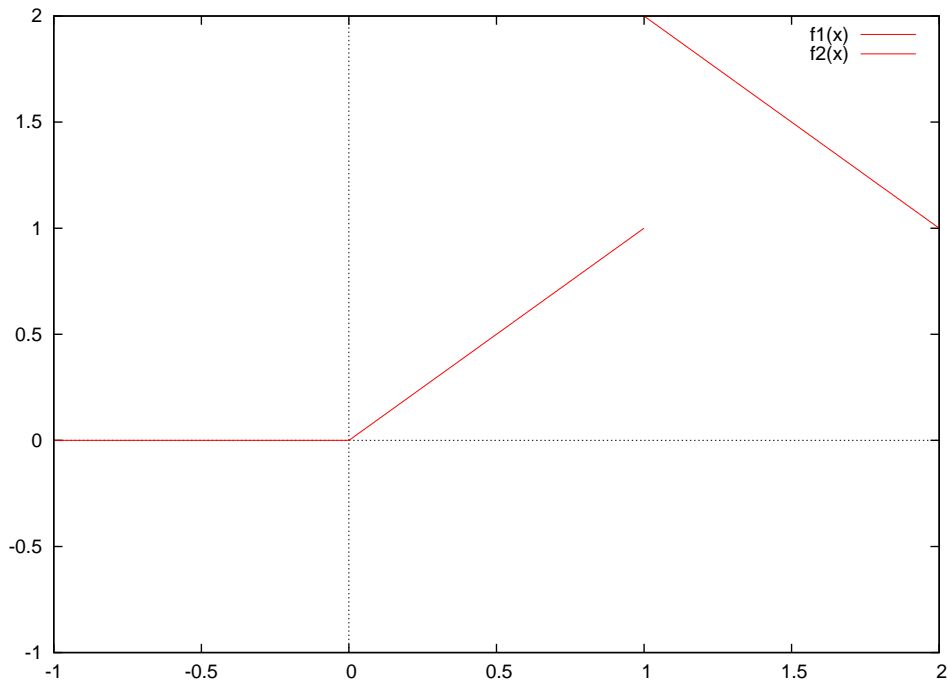
$$f(x) = \begin{cases} \frac{1}{2} \cdot (x + |x|) & x \leq 1 \\ 3 - x & x > 1 \end{cases}$$

môžeme nakresliť pomocou príkazov

```
set zeroaxis
set xrange [-1:2]
set yrange [-1:2]
set samples 1000
f1(x) = x<=1 ? (x+abs(x))/2. : 1/0
f2(x) = x>1 ? 3-x : 1/0
plot f1(x), f2(x) linetype 1
```

Výsledok vidíme na obrázku číslo 36. Ten istý výsledok dosiahneme aj príkazmi

```
set zeroaxis
set samples 3001
set xrange [-1:2]
set yrange [-1:2]
f(x) = x<1 ? (x+abs(x))/2. : x>1 ? 3-x : 1/0
plot f(x)
```



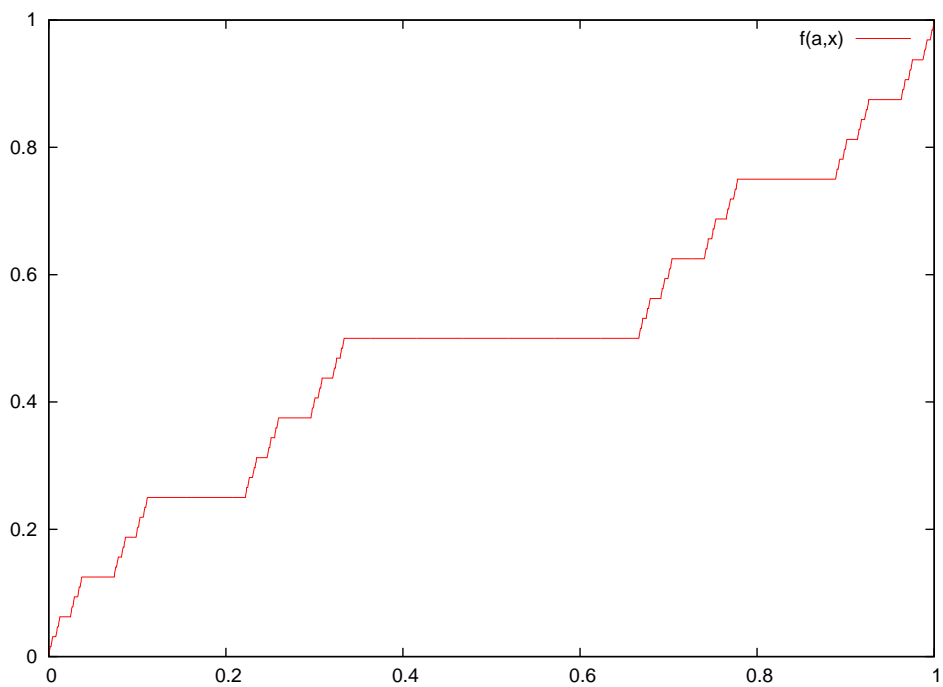
Obr. 36

Ukážeme si, ako môžeme znázorniť polygonálne aproximácie Cantorovej funkcie (the devil's staircase). Vytvoríme si súbor s názvom `cantor.plt` (v nejakom ASCII editore) s nasledujúcim obsahom

```
set samples 2000; set xrange [0:1]; set yrange [0:1]
start(x)=x<=0?0:0<=x&&x<=1?x:1
f(a,x)=a==0?start(x):x<=1/3.?f(a-1,3*x)/2:f(a-1,3*x-2)/2+0.5
plot f(a,x)
a=a+1; pause -1
load 'cantor.plt'
```

a potom v `gnuplot` napíšeme príkazy
`a=0;load 'cantor.plt'`

Výsledok vidíme na obrázku číslo 37.



Obr. 37

Teraz si ukážeme, ako môžeme vyšrafovať elementárnu oblasť

$$a \leq x \leq b$$

$$f(x) \leq y \leq g(x)$$

Vyberieme si niečo jednoduché, napríklad

$$0 \leq x \leq 1$$

$$x^3 \leq y \leq x^2$$

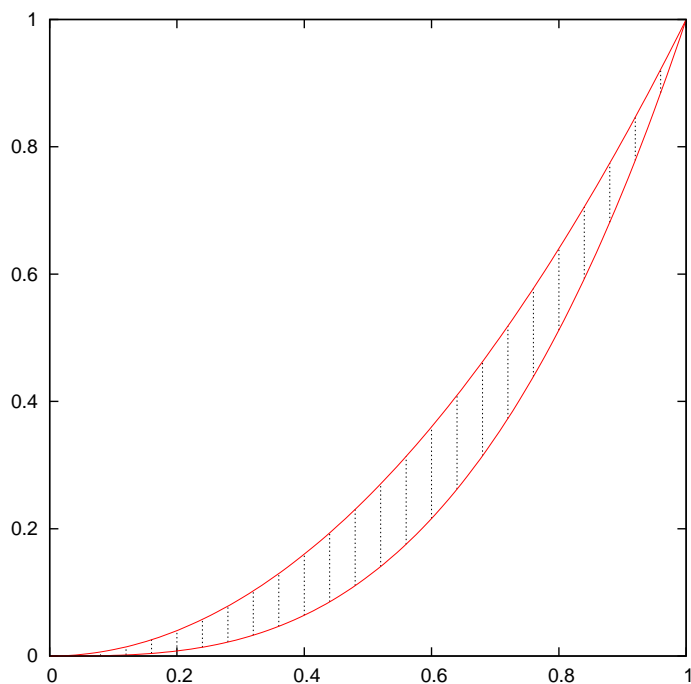
Najskôr si vytvoríme súbor s názvom `re.plt` (v nejakom ASCII editore) s nasledujúcim obsahom

```
i = i+1; x(i)=a+i*(b-a)*n**(-1)
set arrow i from x(i), f(x(i)) to x(i), g(x(i)) nohead linetype 0
if (i<n) reread
```

a potom v `gnuplot` napíšeme príkazy

```
unset key
set size square
a=0; b=1
f(x)=x**3; g(x)=x**2
n=25; i=0
load 're.plt'
plot [0:1] [0:1] f(x), g(x) linetype 1
```

Výsledok vidíme na obrázku číslo 38.



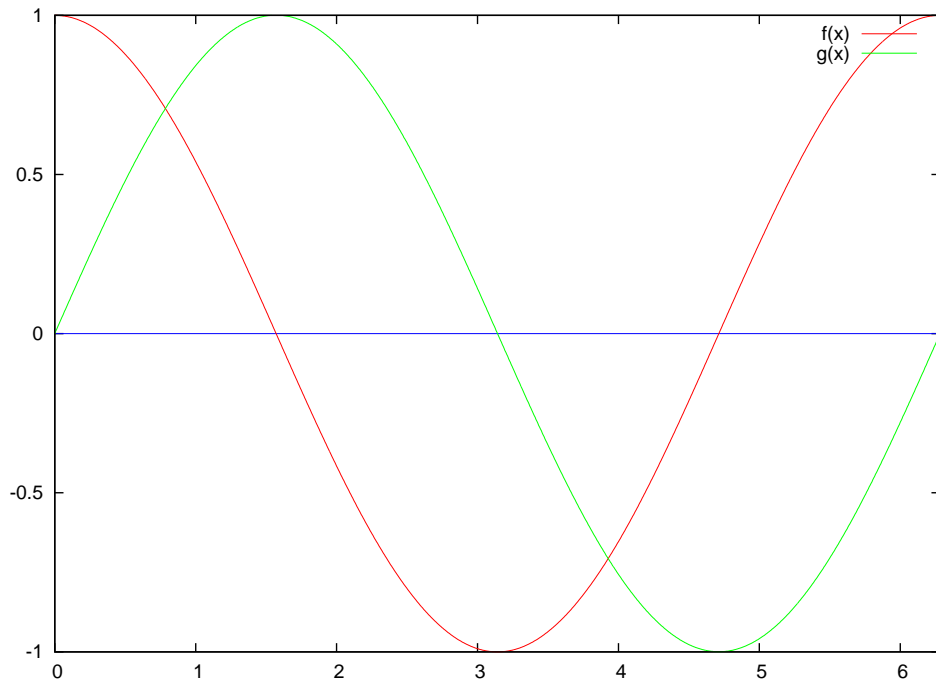
Pre danú funkciu $y = f(x)$ môžeme nakresliť jej integrál (ako funkciu hornej hranice), t. j. graf funkcie

$$g(x) = \int_a^x f(t) dt.$$

Pre funkciu $y = \cos x$ a bod $a = 0$ použijeme príkazy

```
set samples 1000
set xzeroaxis linetype 3
f(x)=cos(x) # sem napíšem predpis funkcie y=f(x)
delta=0.001 # krok, ktorý sa použije pri výpočte aproximácie integrálu
a=0 # dolná hranica integrálu (konštanta)
integral(x,y)= x>y ? 0 : integral(x+delta,y)+delta*f(x)
g(x)= a<x ? integral(a,x) : -integral(x,a)
plot[0:2*pi] [-1:1] f(x), g(x)
```

Výsledok vidíme na obrázku číslo 39.

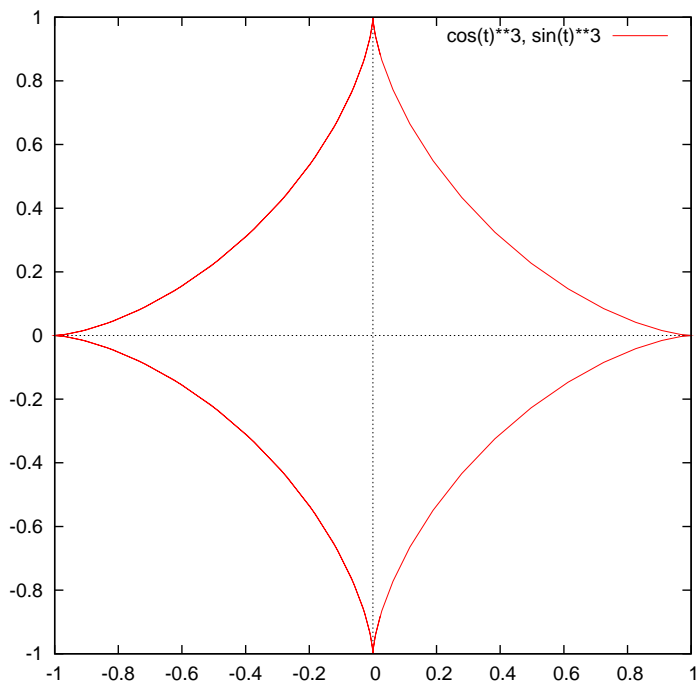


Obr. 39

KRIVKY DANÉ PARAMETRICKY

Asteroidu $\vec{r} = \cos^3 t \cdot \vec{i} + \sin^3 t \cdot \vec{j}$ nakreslíme pomocou príkazov
`set parametric; set size square; set zeroaxis; plot cos(t)**3,sin(t)**3`

Výsledok vidíme na obrázku číslo 40.

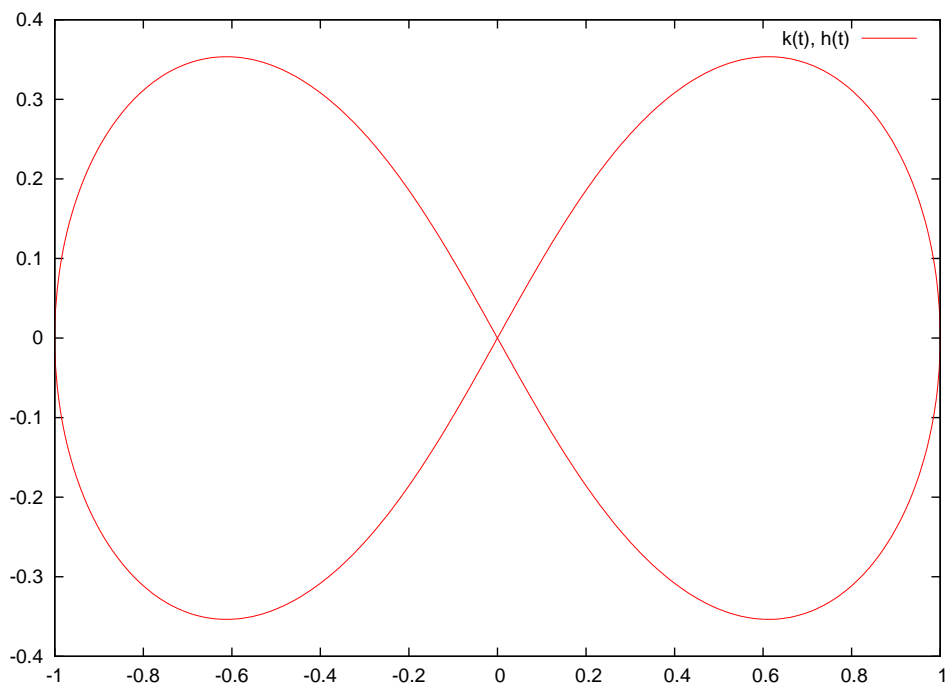


Obr. 40

Lemniskátu nakreslíme pomocou príkazov

```
set parametric; set samples 1000
f(t)=t*(1+t**2)/(1+t**4); g(t)=t*(1-t**2)/(1+t**4)
h(t)= t<1 ? g(t) : g(t-2); k(t)= t<1 ? f(t) : -f(t-2)
set trange [-1:3]; plot k(t),h(t)
```

Výsledok vidíme na obrázku číslo 41.



Obr. 41

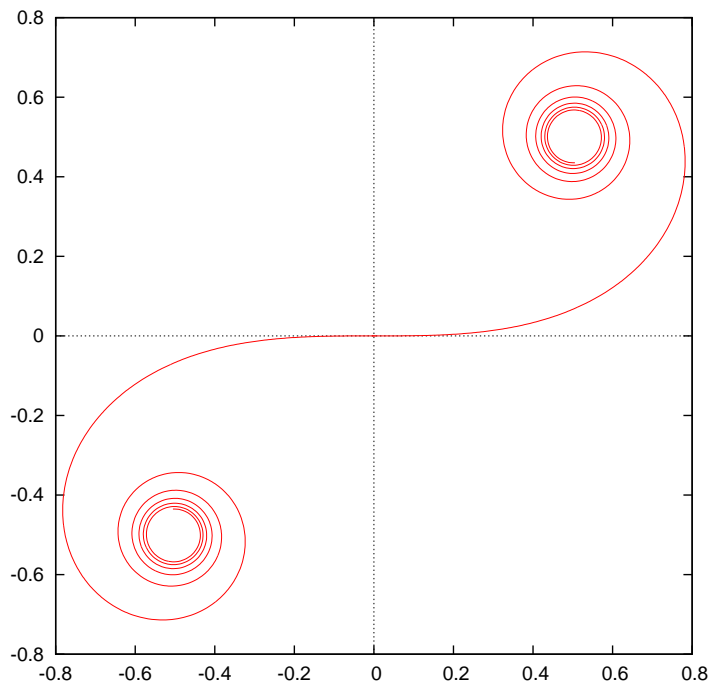
Klotoida má parametrické vyjadrenie

$$\vec{r} = \int_0^t \cos \frac{\pi\theta^2}{2} d\theta \cdot \vec{i} + \int_0^t \sin \frac{\pi\theta^2}{2} d\theta \cdot \vec{j}.$$

Nakreslíme ju pomocou príkazov

```
unset key
set samples 1000
set size square
set zeroaxis
f0(x)=cos(pi/2.*x**2)
f1(x)=sin(0.5*pi*x**2)
integral0(x,y)=x>y?0:integral0(x+delta,y)+delta*f0(x)
integral1(x,y)=x>y?0:integral1(x+delta,y)+delta*f1(x)
g0(x)=a<x?integral0(a,x):-integral0(x,a)
g1(x)=a<x?integral1(a,x):-integral1(x,a)
delta=0.005; a=0
set param; set trange [0:4.9]
plot g0(t),g1(t),-g0(t),-g1(t) linetype 1
```

Výsledok vidíme na obrázku číslo 42.



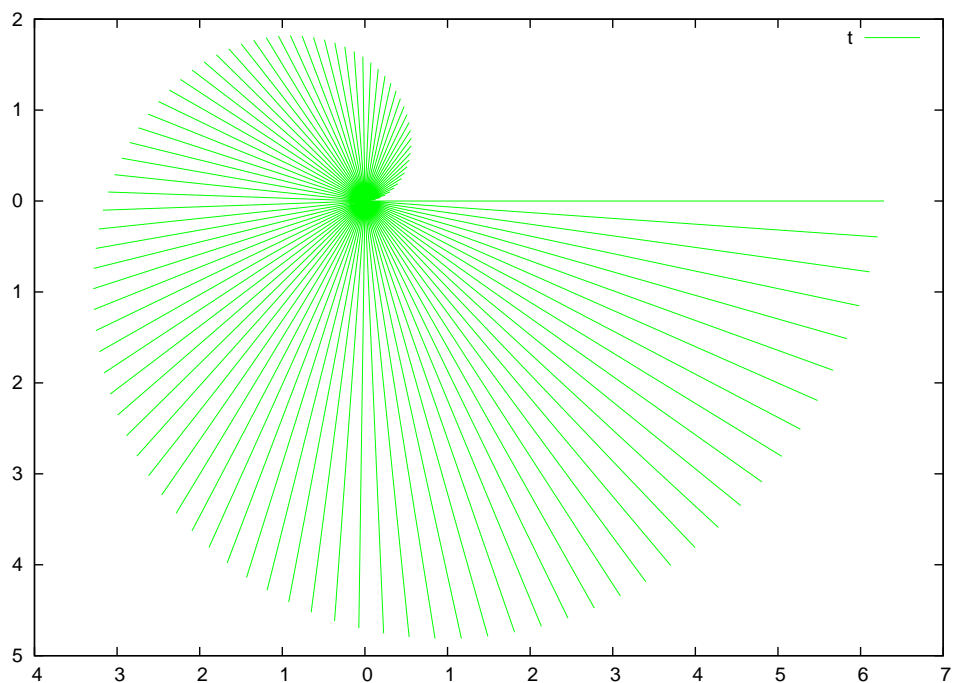
Obr. 42

POLÁRNE SÚRADNICE

Nakreslíme si špirálu v polárnych súradniciach pomocou príkazov

```
set polar
set function style impulses
plot t linetype 2
```

Výsledok vidíme na obrázku číslo 43.



Obr. 43

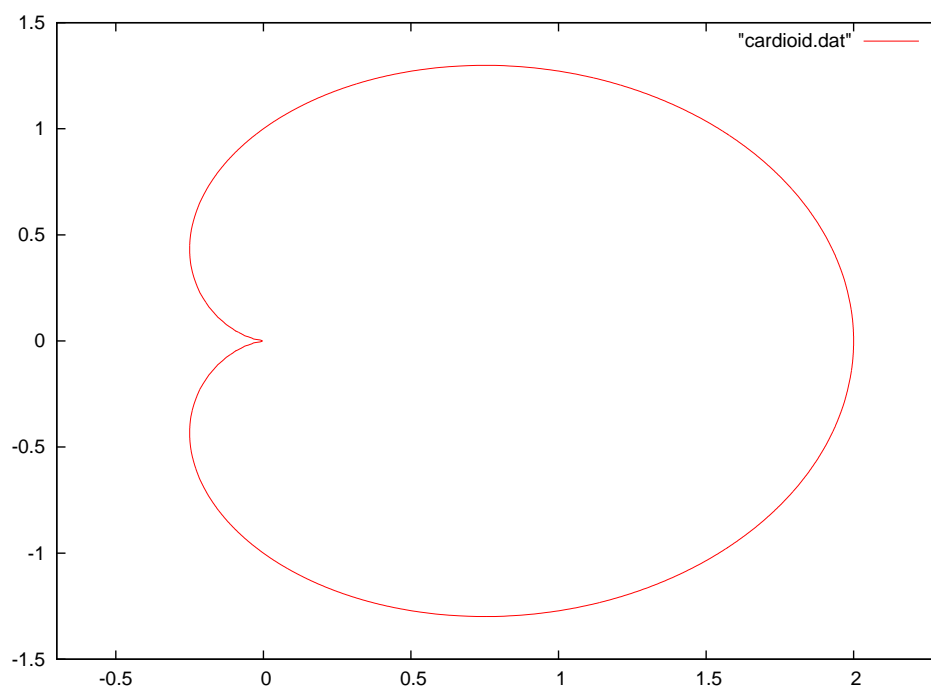
Nakreslíme si kardioidu, ktorá je daná implicitnou rovnicou

$$(x^2 + y^2 - x)^2 = x^2 + y^2.$$

Použijeme príkazy

```
set isosamples 1000
set xrange [-0.7:2.3]
set yrange [-1.5:1.5]
set contour base
set cntrparam levels discrete 0.0
unset surface
set term table
set output "cardioid.dat"
splot (x**2+y**2-x)**2-x**2-y**2
set output
set term windows
plot "cardioid.dat" with lines
```

Výsledok vidíme na obrázku číslo 44.



Obr. 44

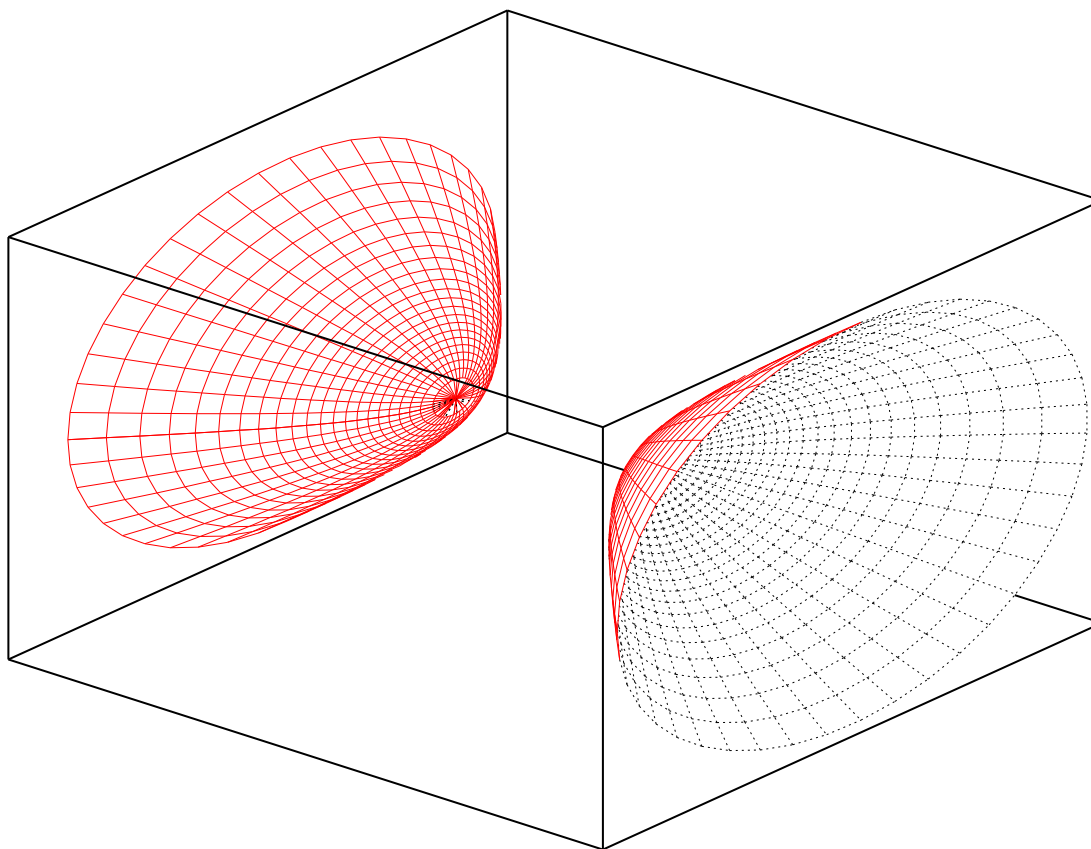
Dvojdielny hyperboloid

$$x^2 - y^2 - z^2 - 1 = 0$$

nakreslíme pomocou príkazov

```
set border 4095
unset key
unset xtics
unset ytics
unset ztics
set parametric
set view 55,220
set isosamples 44,70
set hidden3d
set urange [0:2*pi]
set vrange [-1:1]
set xrange [-3:3]
set yrange [-3:3]
set zrange [-3:3]
f(u,v)=sqrt(1/v**2-1)*cos(u)
g(u,v)=sqrt(1/v**2-1)*sin(u)
splot 1/v, f(u,v), g(u,v) linetype 0
```

Výsledok vidíme na obrázku číslo 45.



Obr. 45

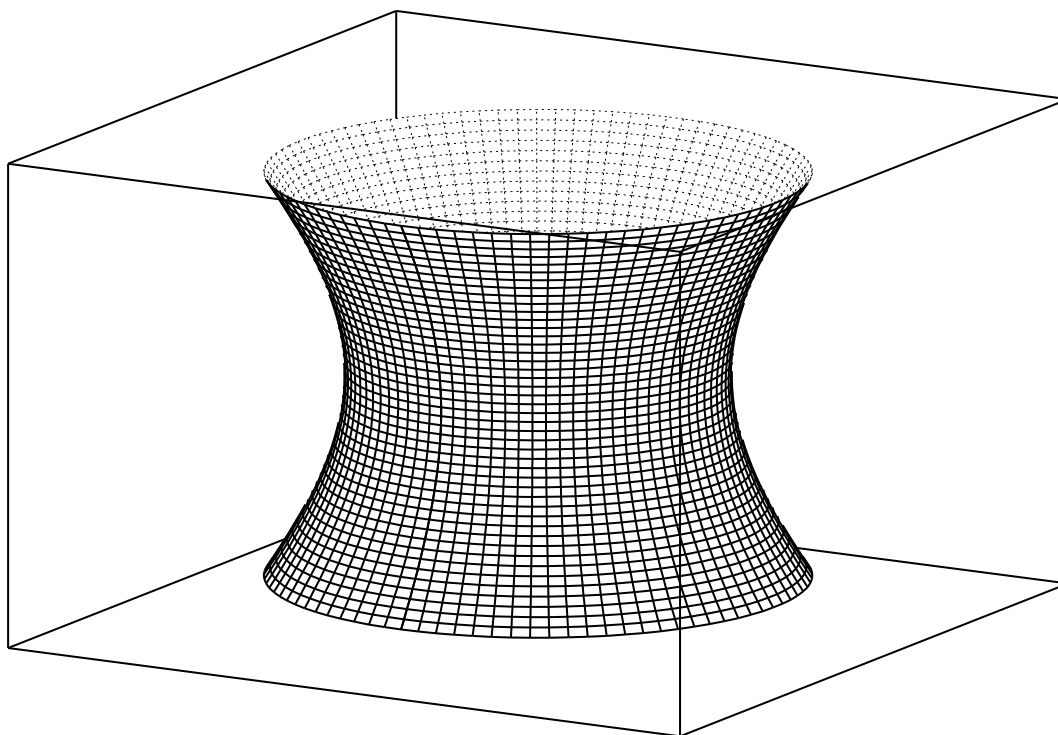
Jednodielny hyperboloid

$$x^2 + y^2 - z^2 - 1 = 0$$

nakreslíme pomocou príkazov

```
set border 4095
unset key
unset xtics
unset ytics
unset ztics
set parametric
set view 70, 120
set isosamples 90, 46
set hidden3d
set urange [0:2*pi]
set vrange [-1:1]
set xrange [-2:2]
set yrange [-2:2]
set zrange [-1.2:1.2]
splot sqrt(1+v**2)*cos(u),sqrt(1+v**2)*sin(u),v linetype -1
```

Výsledok vidíme na obrázku číslo 46.

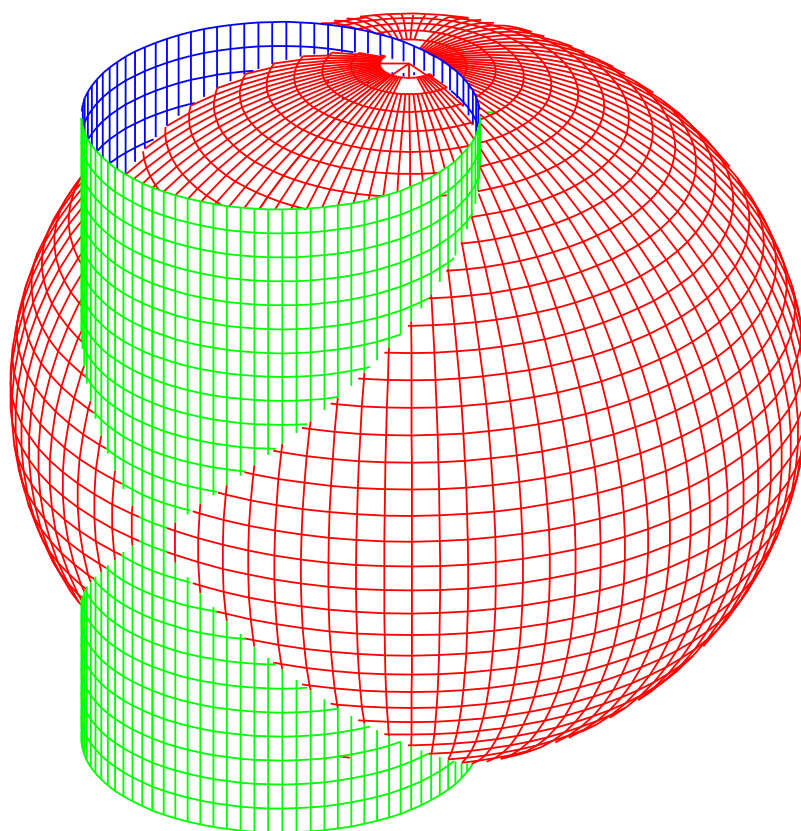


Obr. 46

Valcová a guľová plocha v jednom obrázku (ilustrácia vzniku Vivianiho krivky).

```
unset border
unset key
unset xtics
unset ytics
unset ztics
set parametric
set hidden3d
set view 45, 130
set isosamples 90,44
set urange [0:2*pi]
set vrange [-pi/2:pi/2]
set xrange [-1.75:1.75]
set yrange [-1.75:1.75]
set zrange [-0.9:1.1]
gx(u,v)=cos(u)*cos(v)
gy(u,v)=sin(u)*cos(v)
gz(u,v)=sin(v)
vx(x)=0.5*(1+cos(x))
vy(x)=0.5*sin(x)
plot gx(u,v),gy(u,v),gz(u,v), vx(u),vy(u),v linetype 2
```

Výsledok vidíme na obrázku číslo 47.



Obr. 47

Teraz si ukážeme, ako možno nakresliť graf funkcie $z = f(x, y)$, ktorá je definovaná na elementárnej

oblasti

$$a \leq x \leq b$$

$$\varphi(x) \leq y \leq \psi(x)$$

Pre funkciu $z = \sin x + \cos y + 1$ definovanú na elementárnej oblasti

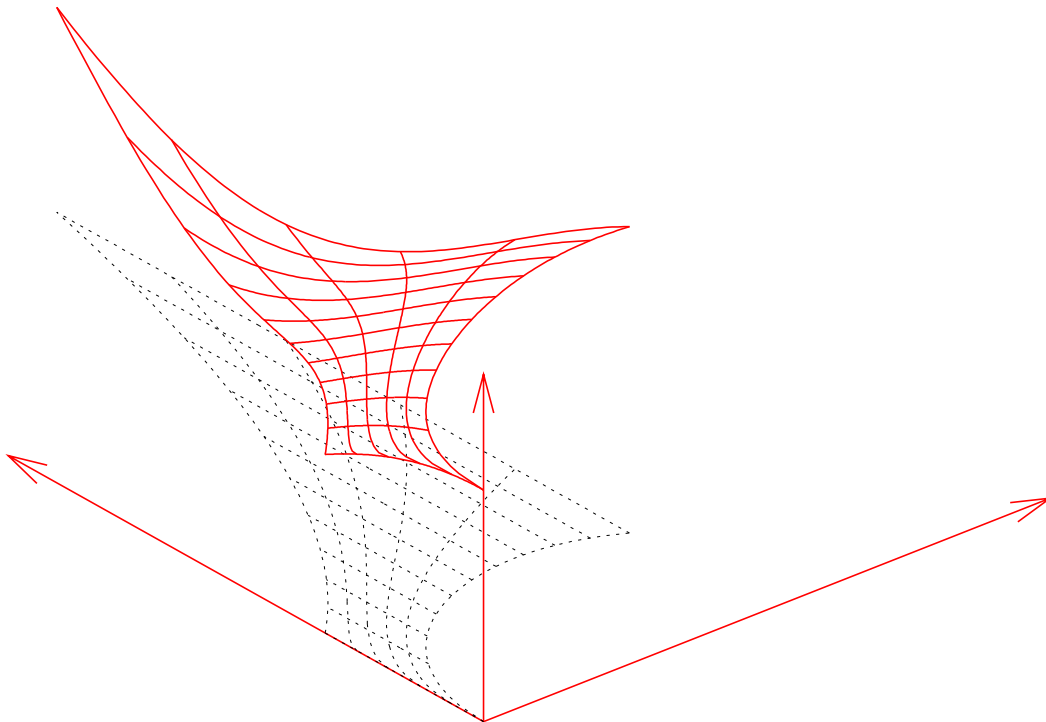
$$0 \leq x \leq \frac{3}{2}$$

$$\sqrt{1 - (1 - x)^2} \leq y \leq e^x$$

(nakreslíme si aj jej definičný obor) použijeme príkazy

```
unset key
unset border
unset xtics
unset ytics
unset ztics
set parametric
set arrow 1 to 3,0,0
set arrow 2 to 0,3,0
set arrow 3 to 0,0,3
set isosamples 11,6
set view 45,320
dolna(x)=sqrt(1-(1-x)**2)
horna(x)=exp(x)
a=0; b=3./2.
f(x,y)=sin(x)+cos(y)+1
mix(x,y)=dolna(x)+(horna(x)-dolna(x))*y
set xrange [0:5]; set yrange [0:5]; set zrange [0:5]
set urange [a:b]; set vrange [0:1]
splot u,mix(u,v),0 linetype 0, u,mix(u,v),f(u,mix(u,v)) linetype 1
```

Výsledok vidíme na obrázku číslo 48.



Obr. 48

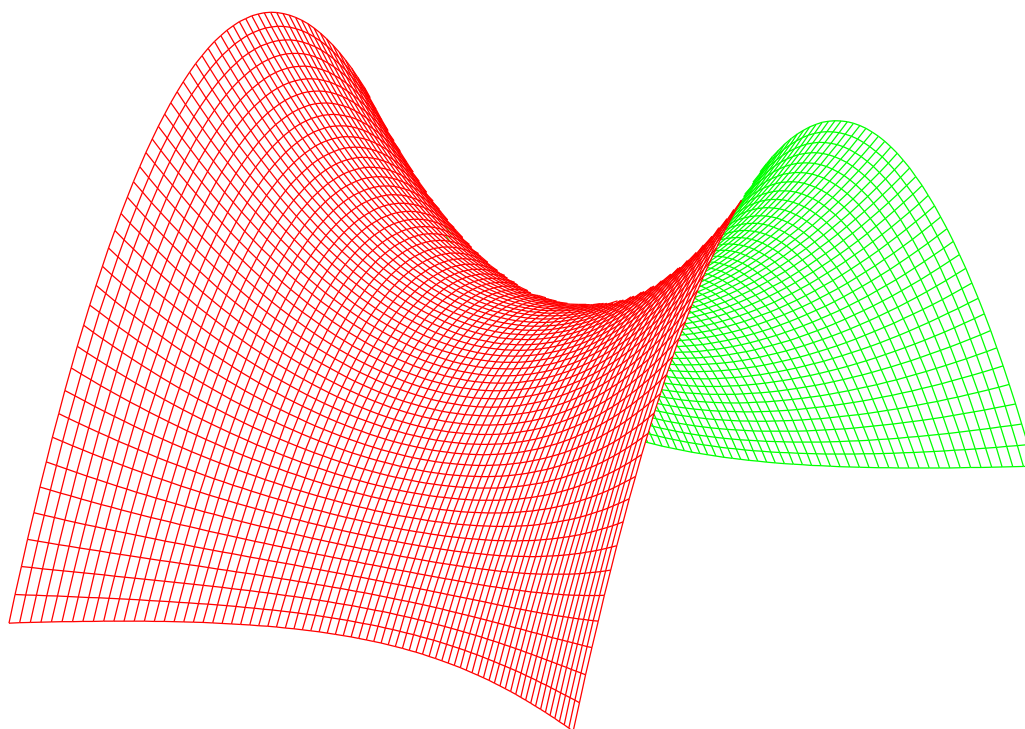
Hyperbolický paraboloid

$$x^2 - y^2 - z = 0$$

nakreslíme pomocou príkazov

```
unset key
unset border
unset xtics
unset ytics
unset ztics
set view 60, 30
set isosamples 75
set hidden3d
a=-1; b=1
set param
set urange [a:b]
set vrange [0:1]
set xrange [-1.5:1.5]
set yrange [-1.5:1.5]
set zrange [-1.2:1.2]
dolna(x)=-sqrt(x*x+1)
horna(x)=sqrt(x*x+1)
f(x,y)=x*x-y*y
mix(x,y)=dolna(x)+(horna(x)-dolna(x))*y
splot u,mix(u,v),f(u,mix(u,v)) linetype 1
```

Výsledok vidíme na obrázku číslo 49.

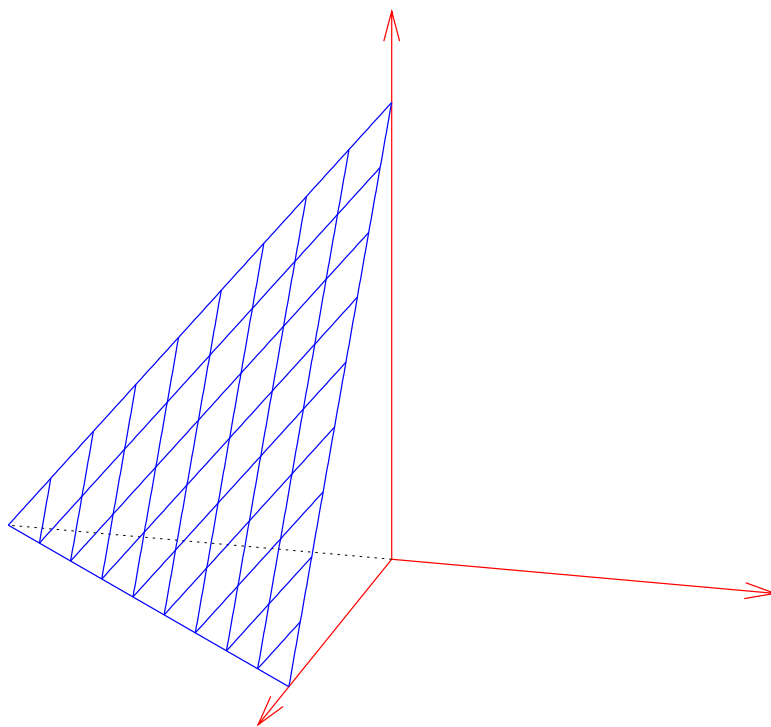


Obr. 49

Časť roviny $x - y + z - 1 = 0$ ležiacu vo štvrtom oktante nakreslíme pomocou príkazov

```
unset key
unset border
unset xtics
unset ytics
unset ztics
set arrow 1 to 1.3,0,0
set arrow 2 to 0,1,0
set arrow 3 to 0,0,1.2
set arrow 4 from 1,0,0 to 0,-1,0 nohead linetype 3
set arrow 5 to 0,-1,0 nohead linetype 0
set parametric; set view 60, 105
set urange [0:1]
set vrange [-1:0]
set xrange [-1.2:1.2]
set yrange [-1.2:1.2]
set zrange [0:1.2]
splot u,v,1-u+v linetype 3
```

Výsledok vidíme na obrázku číslo 50.



Obr. 50

Nakreslíme si graf funkcie

$$f(x, y) = \max \left(\max(x, y), \varphi \left(\frac{x + y}{2} \right) \right),$$

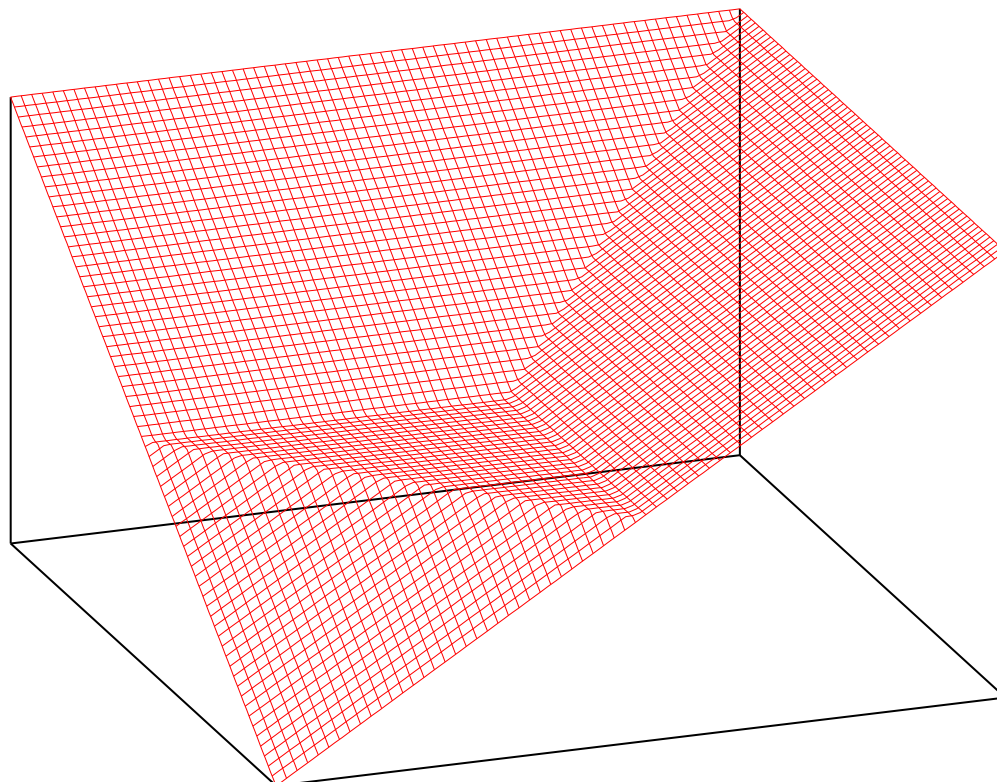
kde

$$\varphi(x) = \begin{cases} 2x & x \leq \frac{1}{2} \\ 1 & \frac{1}{2} < x \leq 1 \\ x & 1 < x \end{cases}$$

Stačí použiť príkazy

```
unset key
unset xtics
unset ytics
unset ztics
set border
set view 60,340
set isosamples 70
f1(x)= x<=0.5 ? 2*x : 1
f2(x)= x<=1 ? f1(x) : x
g(x,y)=max(max(x,y),f2((x+y)/2.))
max(x,y)= x<=y ? y : x
splot [0:2] [0:2] [0:2] g(x,y)
```

Výsledok vidíme na obrázku číslo 51.

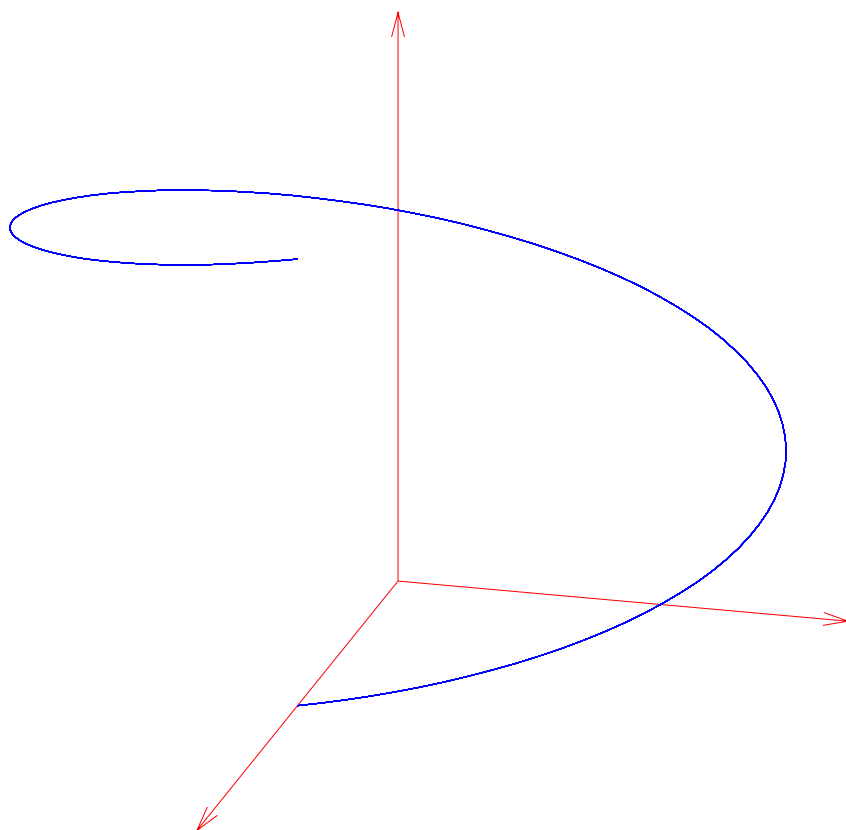


Obr. 51

Skrutkovicu nakreslíme pomocou príkazov

```
unset key
unset border
unset xtics
unset ytics
unset ztics
set arrow 1 to 2,0,0
set arrow 2 to 0,1.2,0
set arrow 3 to 0,0,8
set view 60, 105
set isosamples 35
set param
set urange [0:2*pi]
set vrange [0:2*pi]
splot cos(u),sin(u),u linetype 3
```

Výsledok vidíme na obrázku číslo 52.

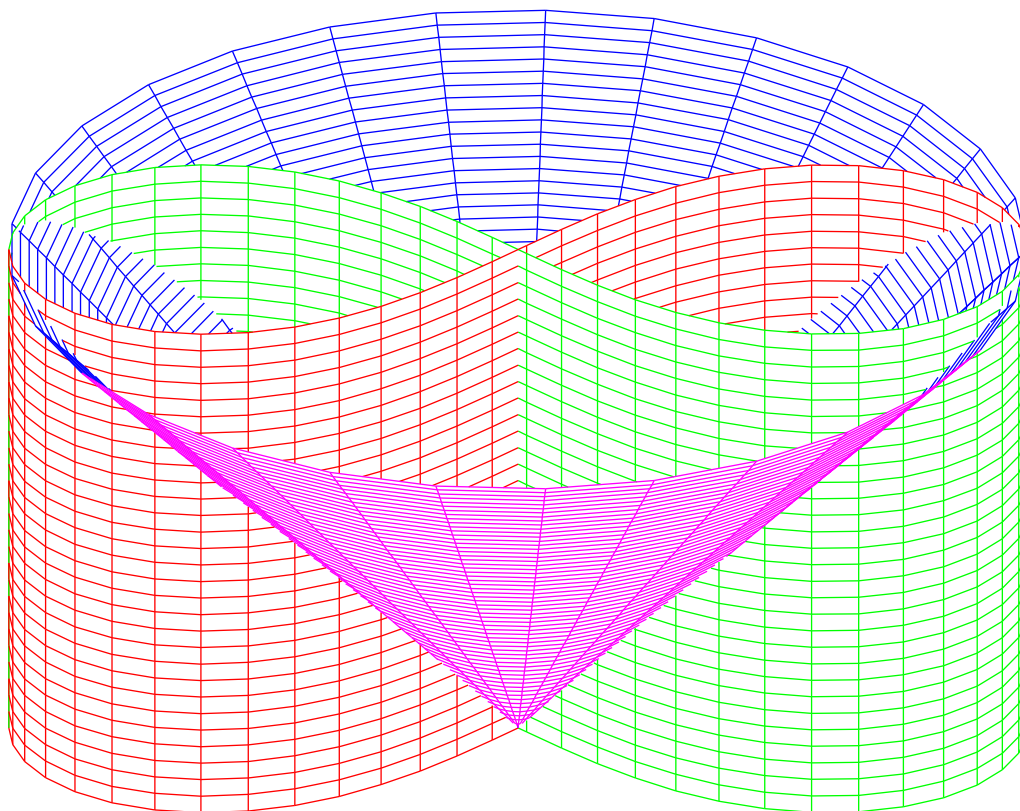


Obr. 52

Prieknik kužeľovej plochy a priamkovej plochy určenej lemniskátou znázorníme pomocou príkazov

```
unset key
unset border
unset xtics
unset ytics
unset ztics
set parametric; set samples 250; set isosamples 60, 30
f(t)=t*(1+t**2)/(1+t**4)
g(t)=t*(1-t**2)/(1+t**4)
h(t)=t<1?g(t):g(t-2)
k(t)=t<1?f(t):-f(t-2)
fx(u,v)=(u+1)/4.*cos(2*pi*v)
fy(u,v)=(u+1)/4.*sin(2*pi*v)
fz(u)=(u+1)/4.
set hidden3d; set view 45,0
plot [-1:3] [0:1] k(u),h(u),v, fx(u,v),fy(u,v),fz(u)
```

Výsledok vidíme na obrázku číslo 53.



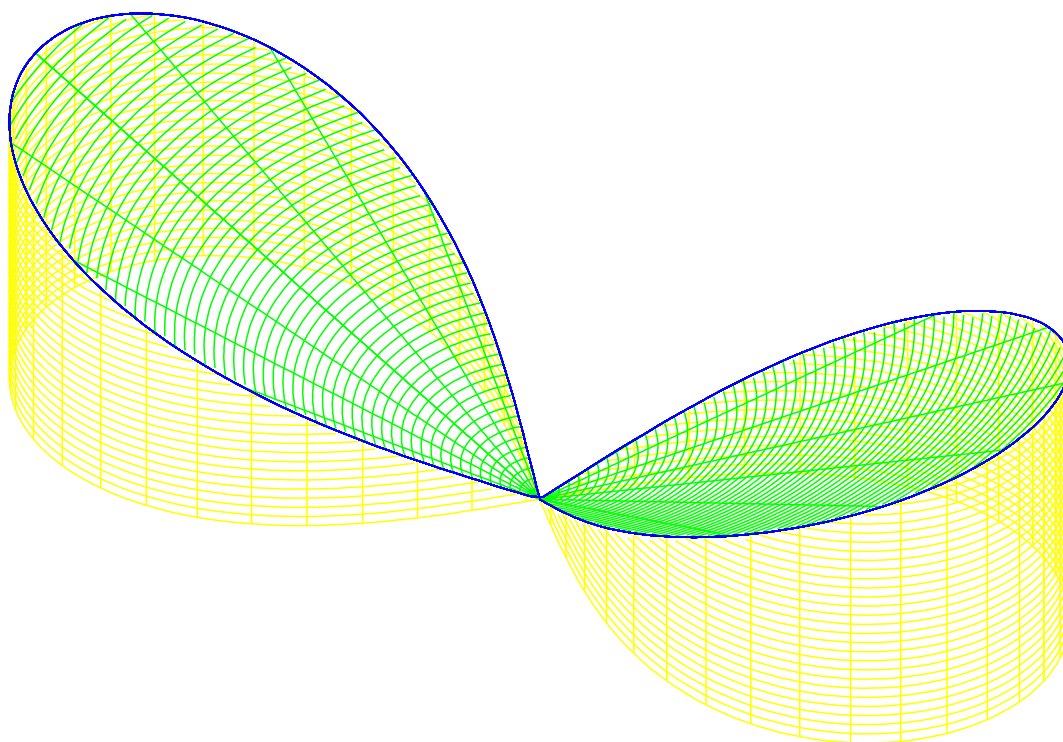
Obr. 53

```

unset key
unset border
unset hidden3d
unset xtics
unset ytics
unset ztics
set parametric
set samples 250, 250
set isosamples 60, 30
set view 20,210
f(t)=t*(1+t**2)/(1+t**4)
g(t)=t*(1-t**2)/(1+t**4)
h(t)=t<1?g(t):g(t-2)
k(t)=t<1?f(t):-f(t-2)
fx(u,v)=(u+1)/4.*cos(2*pi*v)
fy(u,v)=(u+1)/4.*sin(2*pi*v)
fz(u)=(u+1)/4.
splot [-1:3] [0:1] k(u),h(u),v**2>k(u)**2+h(u)**2?1/0:v lt 6,\
fx(u,v),fy(u,v),(fx(u,v)**2+fy(u,v)**2)**2>fx(u,v)**2-fy(u,v)**2?1/0:fz(u) lt 2,\
k(u),h(u),sqrt(k(u)**2+h(u)**2) lt 3

```

dostaneme inú pohľad na tento prienik. Výsledok vidíme na obrázku číslo 54.

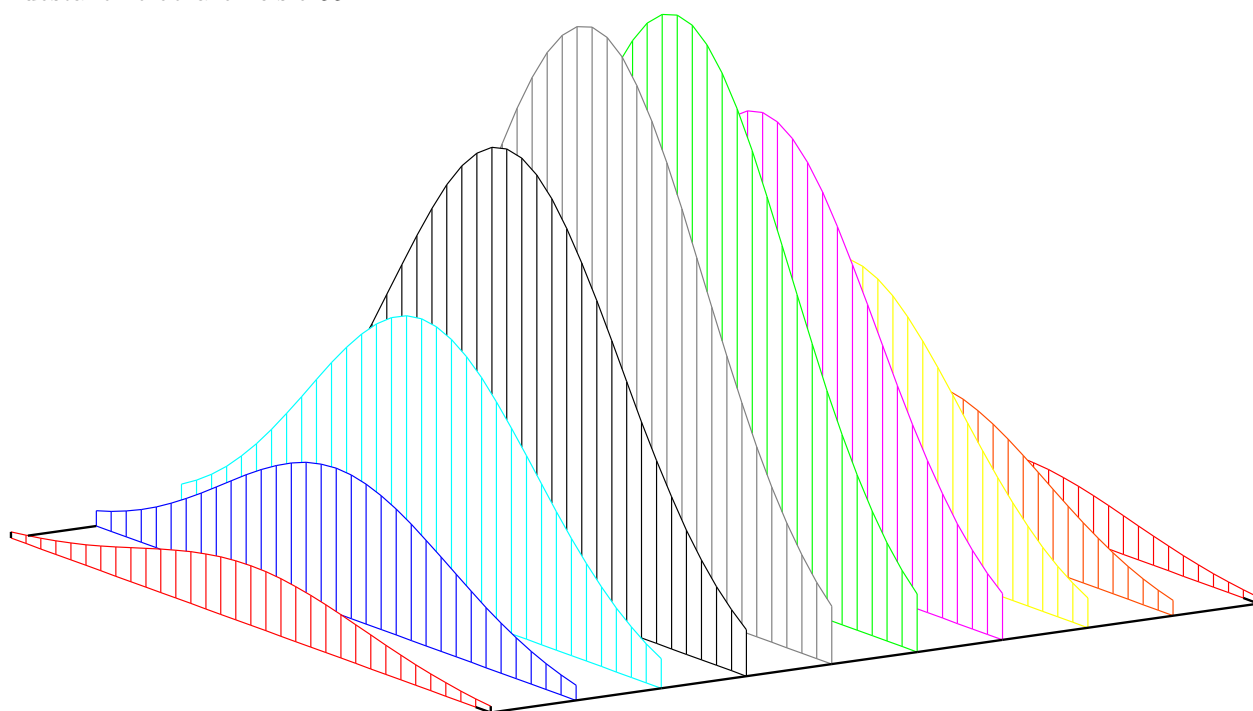


Obr. 54

Pomocou príkazov

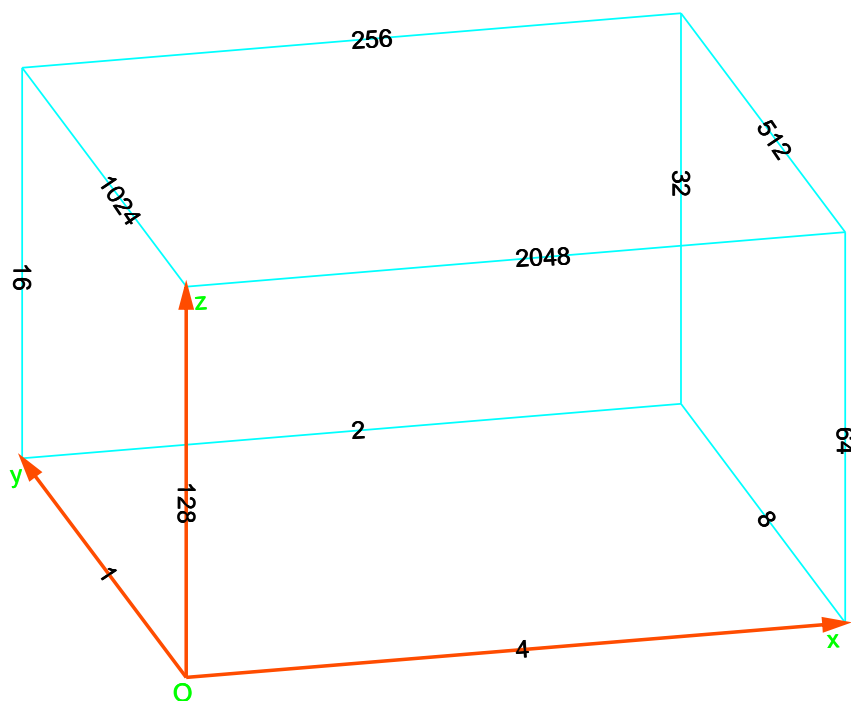
```
unset xtics; unset ytics; unset ztics
unset key; set border 15
set parametric; set hidden3d
set isosamples 2,33; set view 70,328
a=-1.5; h=1/3.
a0=a; a=a+h
a1=a; a=a+h
a2=a; a=a+h
a3=a; a=a+h
a4=a; a=a+h
a5=a; a=a+h
a6=a; a=a+h
a7=a; a=a+h
a8=a; a=a+h
a9=a
f(x,y) = exp(-(x**2+y**2))
splot [0:1] [-1.5:1.5] [-1.5:1.5] [-1.5:1.5] [0:1]\
a0, v, (u<1) ? 0 : f(a0,v), \
a1, v, (u<1) ? 0 : f(a1,v), \
a2, v, (u<1) ? 0 : f(a2,v), \
a3, v, (u<1) ? 0 : f(a3,v), \
a4, v, (u<1) ? 0 : f(a4,v), \
a5, v, (u<1) ? 0 : f(a5,v), \
a6, v, (u<1) ? 0 : f(a6,v), \
a7, v, (u<1) ? 0 : f(a7,v), \
a8, v, (u<1) ? 0 : f(a8,v), \
a9, v, (u<1) ? 0 : f(a9,v)
```

dostaneme obrázok číslo 55.



Obr. 55

Jednotlivé hrany rámcika v priestorovom grafe sú očíslované mocninami dvojky (podobne ako to bolo v prípade rovinných grafov). Kódovanie hrán ilustruje nasledujúci obrázok



Obr. 56

Napríklad rámcík so všetkými hranami nakreslíme príkazom `set border 4095`.

Čísla priradené vykresľovaným hranám totiž musíme sčítať, teda

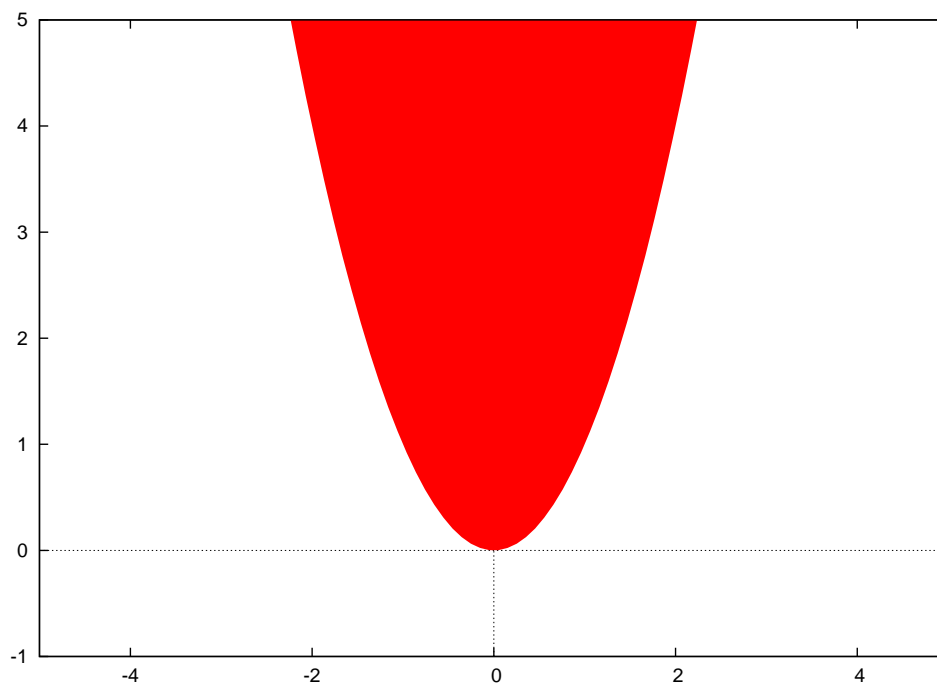
$$4095 = 1 + 2 + 4 + 8 + 16 + 32 + 64 + 128 + 256 + 512 + 1024 + 2048.$$

Príkaz filledcurve.

Pomocou príkazov

```
unset key; set zeroaxis; plot [-5:5] [-1:5] x**2 with filledcurve
```

nakreslíme nasledujúci obrázok

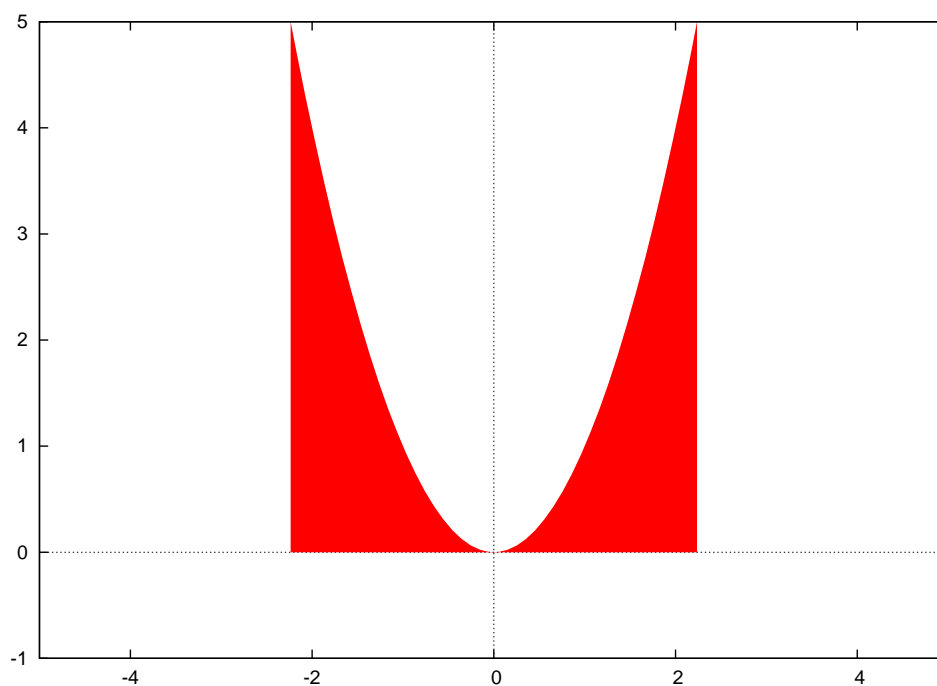


Obr. 57

Malú modifikáciu dostaneme pomocou príkazov

```
unset key;set zeroaxis; plot [-5:5] [-1:5] x**2 with filledcurve y1=0
```

Výsledok vidíme na obrázku číslo 58.

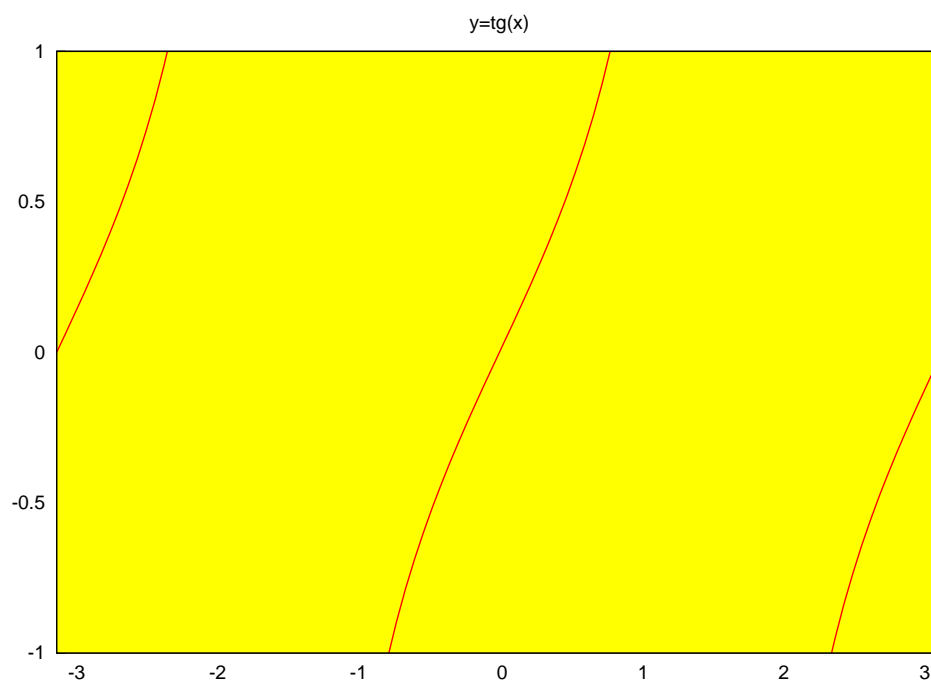


Obr. 58

Ďalšie využitie tohto príkazu dosiahneme pomocou príkazov

```
set title "y=tg(x)"  
unset key  
plot [-pi:pi] [-1:1] -1 with filledcurve x2 lt 15,tan(x) lt 1 lw 2
```

Výsledok vidíme na obrázku číslo 59.



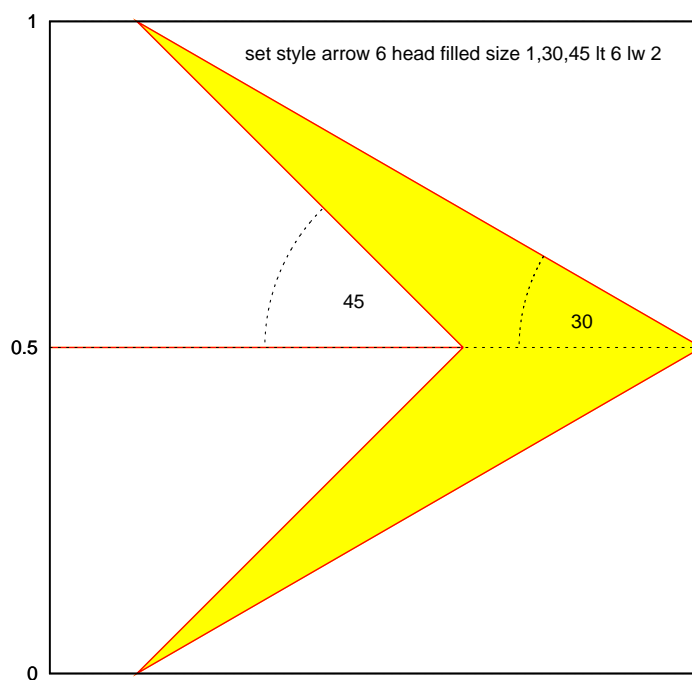
Obr. 59

Orientované úsečky. Podstatným spôsobom sa rozšírili možnosti ukončovania úsečiek šípkami. Jednotlivé možnosti vidíme na obrázku číslo 60.



Obr. 60

Základné parametre šípky sú názorne ilustrované na obrázku číslo 61.



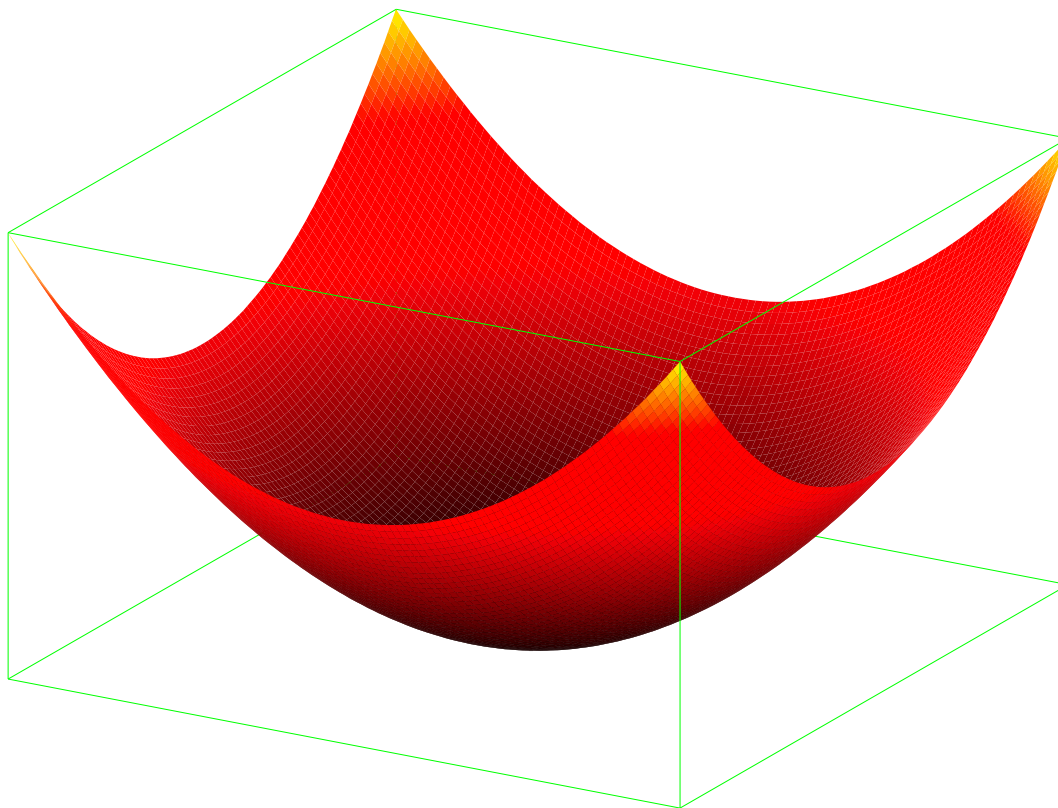
Obr. 61

Štýl pm3d. Na záver si ukážeme niektoré možnosti tohto štýlu pri kreslení priestorových grafov.

Pomocou príkazov

```
set border 4095 lt 2  
set isosamples 100  
set noxtics  
set noytics  
set noztics  
set pm3d solid  
set style line 100 lt 0 lw 0  
unset hidden3d  
unset surf  
unset colorbox  
set palette model XYZ rgbformulae 3,23,36  
splot x*x+y*y
```

nakreslíme nasledujúci obrázok

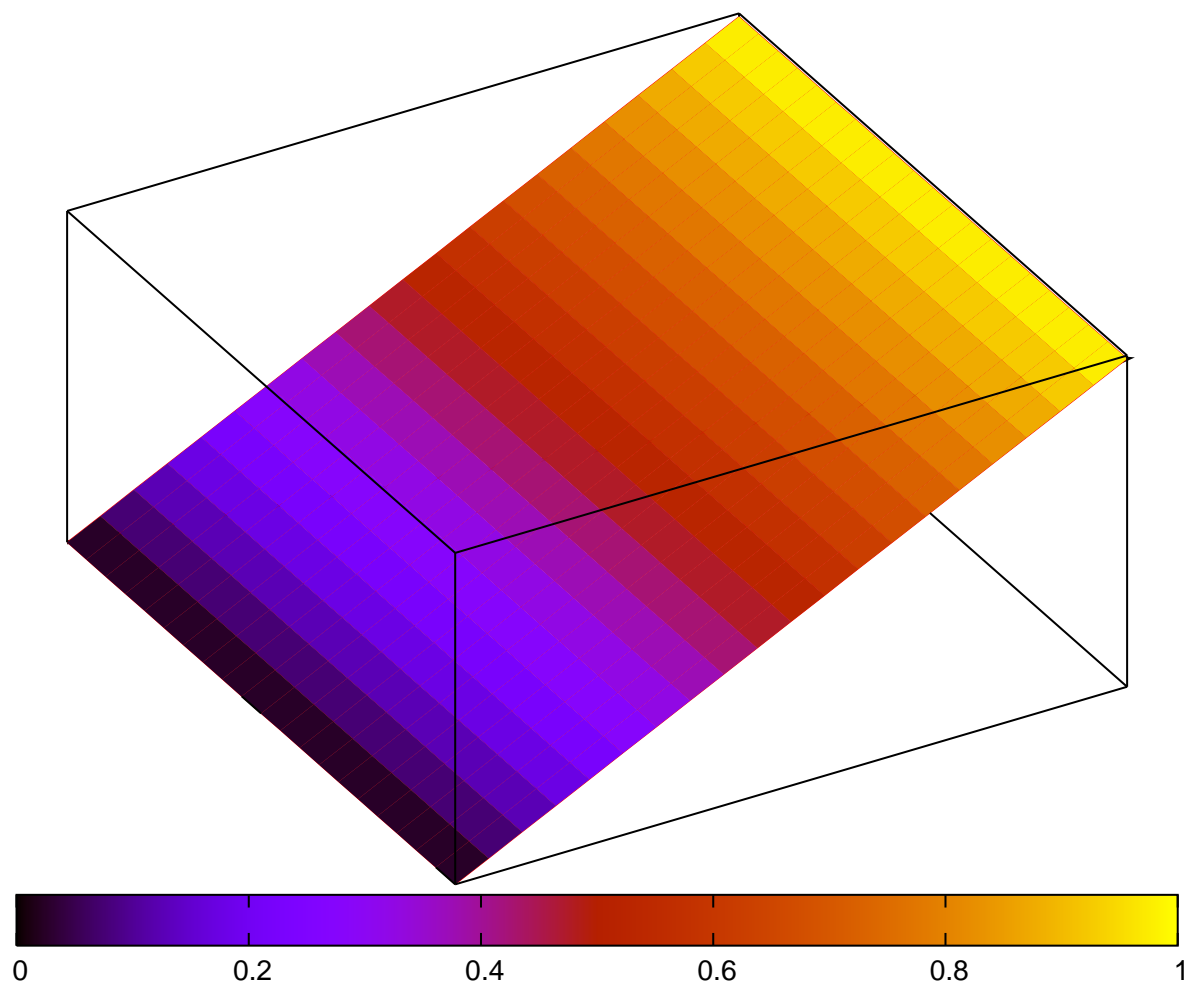


Obr. 62

Pomocou príkazov

```
unset key
set border 4095
set noxtics; set noytics; set noztics
set isosamples 21
set pm3d
set colorbox horiz user origin 0.1,0.025 size 0.8,0.05
set style arrow 3 head empty size 1,30,45 lt -1 lw 2
set view 40,60
set arrow 1 to 1,1,1 as 3
set hidden3d
set xrange [0:1]; set yrange [0:1]; set zrange [0:1.01]
splot y
```

nakreslíme nasledujúci obrázok

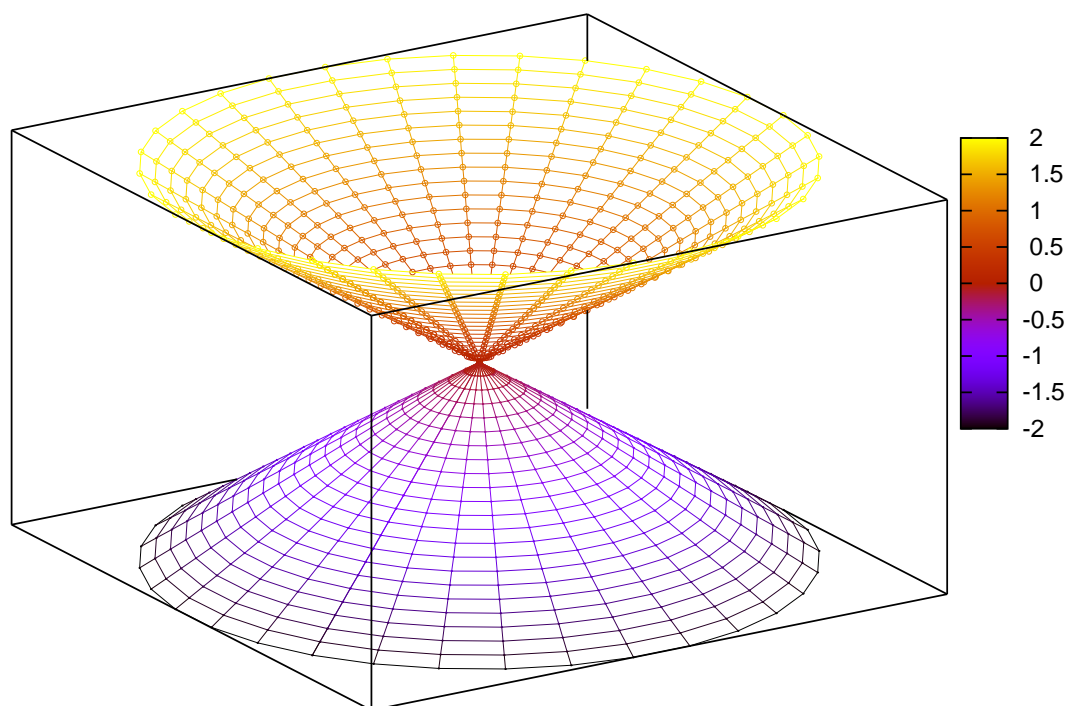


Obr. 63

Pomocou príkazov

```
unset key
set isosamples 23,33
set hidden3d
unset xtics
unset ytics
unset ztics
set border 4095
set view 61, 148
set zeroaxis
set pal maxcolors 0
set surface
set format z
set palette
set param
set urange [0:2]
set vrange [0:2*pi]
set zrange [-2:2]
x(u,v)=u*cos(v)
y(u,v)=u*sin(v)
splot x(u,v+0.5),y(u,v+0.5),u w lp palette pt 6 ps 0.5,\
x(u,v+0.5),y(u,v+0.5),-u w lp palette pt 6 ps 0.1
```

nakreslíme nasledujúci obrázok



Obr. 64

S využitím príkazu `reread` môžeme zautomatizovať vytvorenie série obrázkov pre animáciu. Slúžia nám na to súbory `anim.plt` a `main.plt`, ktoré vytvoríme v nejakom ASCII editore:

```
# anim.plt
i=0
n=13
load 'main.plt'

# main.plt
set print 'anim.plt'
print "set term post eps color"
if (i<10) print "set output '0",i,".ps'"
if (i>9) print "set output '",i,".ps'"
print "unset key"
print "unset xtics"
print "unset ytics"
print "unset ztics"
print "unset border"
print "set hidden3d"
print "posun=15"
print "a=7+",i,"*posun"
print "set view 60, a-floor(a/360)*360"
print "splot x*x-y*y"
load 'anim.plt'
i=i+1
if (i<=n) reread
if (i>n) set print 'anim.plt';\
print "i=", 0; print "n=",n;\
print "load 'main.plt'"
```

Po otvorení súboru `anim.plt` v `gnuplote` sa vygeneruje séria obrázkov

00.ps, 01.ps, ... , 13.ps.

Animáciu môžeme potom vytvoriť napríklad programom `pdfTEX`.⁸

⁸<http://www.fi.muni.cz/~xholecek/tex/pdfanim.xhtml>

Náš výklad ukončíme domácí úlohou:

```
f(x,y)=(x*x-1)**2+(x*x*y-x-1)**2
set isosamples 500, 350
set noxtics
set nokey
set noborder
set noytics
set noztics
set hidden3d
set view 25,40
set xrange [-2.2:2.2]
set yrange [-10:10]
set zrange [-10:0.5]
set pm3d solid
set style line 100 lt 0 lw 0
unset surf
unset colorbox
set palette model XYZ functions 0.05+gray**0.35, gray**0.5, gray**0.8
splot f(x,y)>10?-10:-f(x,y)
```

NÁZOV: GNUPLOT

AUTOR: Jozef Doboš

VYDALA: FEI TU v Košiciach — Edícia vysokoškolských učebníc

POČET STRÁN: 54

VYDANIE: prvé

SADZBA: elektronická, programom $\mathcal{A}\mathcal{M}\mathcal{S}\mathcal{T}\mathcal{E}\mathcal{X}$

ISBN 80-8073-637-5