

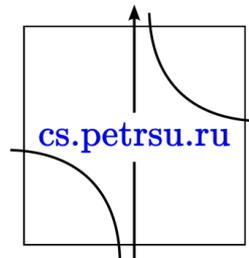
Визуальное представление математических объектов

Введение

Глава №0

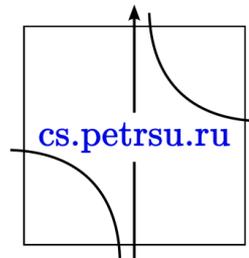
Основное внимание:

- Как следует содержательно конструировать и оформлять график?
- Изучение инструментов, которые позволят вам реализовать ваши "графические" идеи в документах.



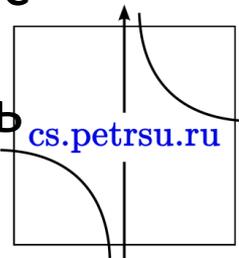
Инструменты:

- интерактивная система рисования функций с командным интерфейсом Gnuplot для составления графиков
- LaTeX как универсальное средство, с помощью которого вы сможете интегрировать полученные графики в документы

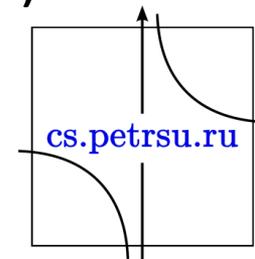


Программа лекций:

1. Общие принципы графического представления данных. (1 час)
2. Особенности графических систем представления данных. (1 час)
3. Gnuplot - общее описание. Команда plot. (1 час)
4. Параметрические функции. Полярная система координат. Команды set parametric и set polar. (1 час)
5. Переменные и встроенные функции. Команда splot. (1 час)
6. Встроенный редактор, операторы, функции, определенные пользователем. (1 час)
7. Файлы данных и файлы команд. Команды работы с файлами команд load, save, reread. Параметры команды plot для работы с файлами данных. Стиль errorbars. (1 час)

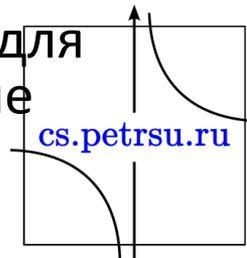


8. Параметры команды set: управление масштабом (autoscale), единицы измерения углов (angles), стрелки (arrow), сетка (dgrid3d, grid), метки (label). (2 часа)
9. Параметры команды set: контуры поверхностей (contour, cntrparam), скрытые линии (hidden3d), изолинии (isosamples). (2 часа)
10. Параметры команды set: оформление осей и разметки (border, format, logscale, tics, ticslevel, x[yz]tics, x[yz]dtics, x[yz]mtics, x[y]label, x[y]tics). (2 часа)
11. Параметры команды set: Общий вид графика (dummy, offset, samples, size, view). (1 час)
12. Параметры команды set: Формат вывода (set terminal Postscript, LaTeX), вывод в файл (output). Интеграция файлов формата Postscript в LaTeX документ. (1 час)
13. Параметры команды set: сферические и цилиндрические координаты (mapping), анимационные графики. (1 час)



Требования к заданиям:

- Отчетное задание должно быть представлено в виде документа в формате Postscript, и соответствующего ему исходного документа в формате LaTeX.
- Отчетный документ состоит из четырех глав, каждая из которых включает в себя ряд рисунков и пояснения к ним.
- Все рисунки должны быть получены с помощью программы gnuplot/python и включены в LaTeX документ непосредственно или в виде eps-файлов.
- Рисунки должны быть включены в документ с помощью окружения figure, подписи под рисунками должны быть сделаны с использованием команды caption.
- Рисунки должны иметь сквозную нумерацию, пояснительный текст должен содержать ссылки на рисунки, путем упоминания их номеров.
- Отчетный документ должен содержать код необходимый для построения графика соответствующего графика (окружение listing).



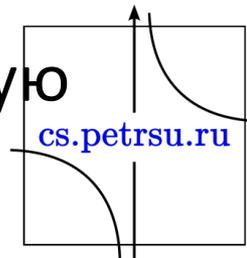
Явные функции одной переменной.

1. Выбрать явную функцию двух переменных. Зафиксировав значения одного из аргументов, получить для нее три изолинии и представить их на одном 2D рисунке. Выбранная функция должна иметь содержательный смысл, а три полученные кривые отражать динамику некоторого процесса. Содержательный смысл процесса должен быть отражен на рисунке. Важные точки полученных кривых должны быть отмечены с помощью arrow и label. Использовать аналитическое представление кривых.
2. Построить график кривой второго порядка, имеющей асимптоту. Асимптота также должна быть представлена на рисунке. Важные точки кривой отметить с помощью arrow и label. Кривая должна быть представлена в виде набора координат.
3. Построить смесь (линейную комбинацию) двух нормальных распределений. Построить график плотности такой смеси и график ее функции распределения. Важные точки плотности отметить с помощью arrow и label. Представление кривых по выбору студента. Одна из кривых должна быть построена в логарифмической системе координат.



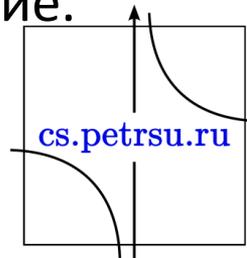
Параметрическое представление кривых и кривые в полярных координатах

1. Построить в полярных координатах графики двух замкнутых кривых на одном рисунке. Кривые должны быть представлены аналитически и иметь природный аналог (содержательный или визуальный).
2. Построить в полярных координатах незамкнутую кривую. Кривая должна быть представлена в виде набора координат и иметь природный аналог (содержательный или визуальный).
3. Построить в декартовых координатах замкнутую кривую, используя ее аналитическое параметрическое представление.



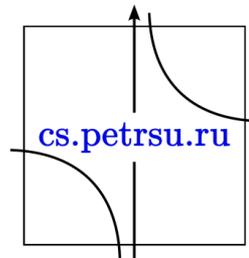
Поверхности

1. Выбрать явную функцию двух переменных и получить ее графическое представление. Выбранная функция должна иметь содержательный смысл, который должен быть отражен на рисунке. Важные точки полученной поверхности должны быть отмечены с помощью `arrow` и `label`. Функция должна быть представлена аналитически.
2. Выбрать явную функцию трех переменных. Получить для нее две изоповерхности и представить их на одном рисунке. Рисунок должен отражать динамику развития некоторого процесса и его содержательную интерпретацию. Важные точки поверхностей должны быть отмечены с помощью `arrow` и `label`. Функция должна быть представлена аналитически.
3. Повторить задание 1 для функции представленной в виде набора координат. Использовать контурное представление. Параметры контура установить самостоятельно.



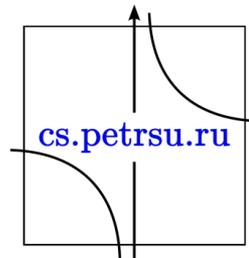
Параметрическое представление поверхностей. Сферические и цилиндрические координаты.

1. Получить графическое представление замкнутой поверхности, используя ее аналитическое параметрическое представление.
2. Получить графическое контурное изображение поверхности, представленной в виде набора сферических координат.
3. Получить представление поверхности, представленной в виде набора ее цилиндрических координат.



Gnuplot

- <http://www.gnuplot.info/download.html>
- Версия для UNIX
- Версия для Windows
- Версия для Mac



Альтернативы:

- Wolfram Mathematica – платная, есть 15-дневная бесплатная версия
- <https://www.wolfram.com/mathematica/>

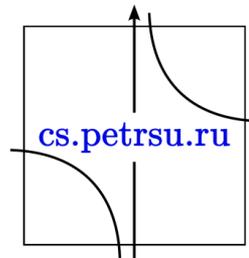
- Mathcad – платная, есть 30-дневная бесплатная версия
- <https://www.mathcad.com/>

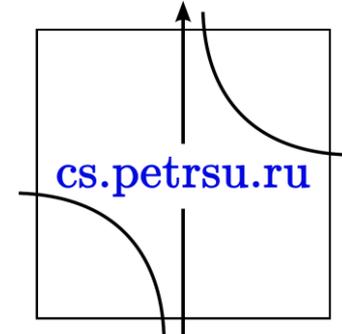
- Matlab – платная, есть 30-дневная бесплатная версия
- <https://www.mathworks.com/>

- Maple – платная, есть 15-дневная бесплатная версия
- <https://www.maplesoft.com/>

- GNU Octave – свободное ПО
- <https://www.gnu.org/software/octave/>

- И т.д.





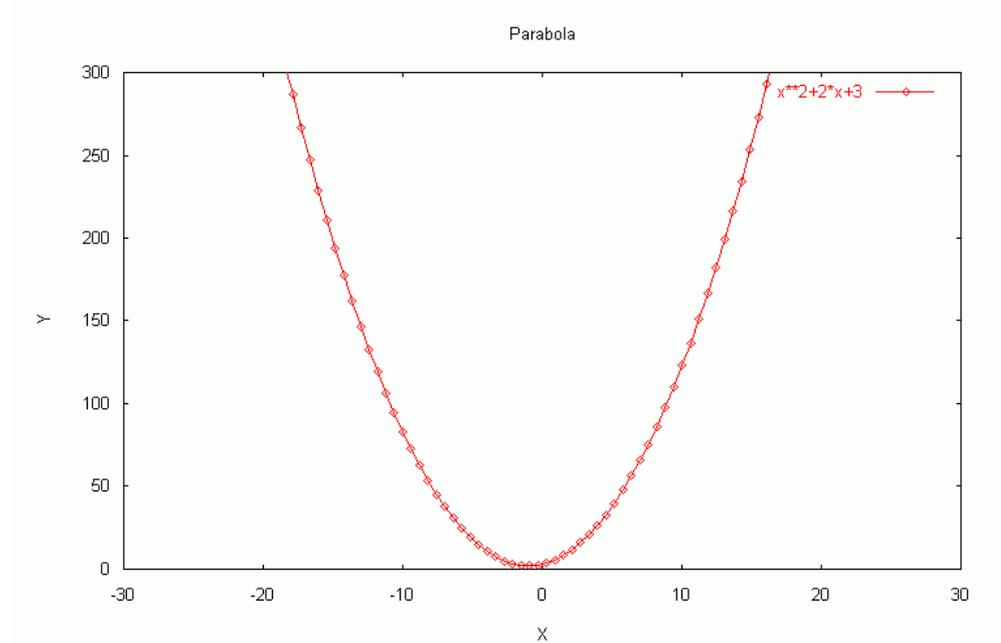
Визуальное представление математических объектов

Примеры построения графиков

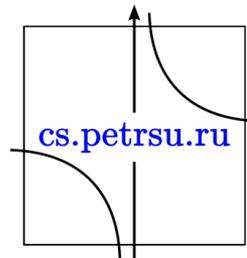
Глава №1

Декартовы координаты. Функция.

Изображаемая фигура: **Парабола**



Команды Gnuplot:
set title "Parabola"
set xlabel "X"
set ylabel "Y"
plot [-30:30] [0:300] $x^{}2+2*x+3$ with lp 1**



Декартовы координаты. Замкнутая.

Изображаемая фигура: **Окружность**

Команды Gnuplot:

set title "Circle"

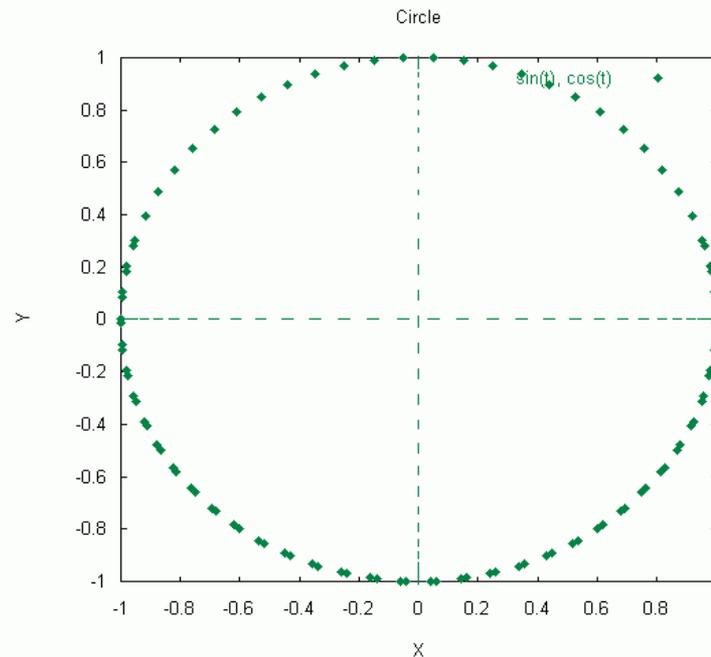
set size ratio 1

set xlabel "X"

set ylabel "Y"

set parametric

plot sin(t),cos(t) with xyerrorbars 10



- В командной строке пишем команду вызова: `gnuplot` и нажимаем клавишу `enter`.
- Получаем следующий результат:

```
chistyak@каппа:~> gnuplot

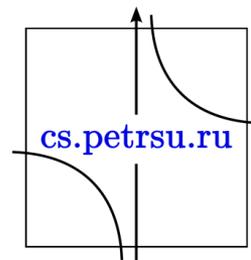
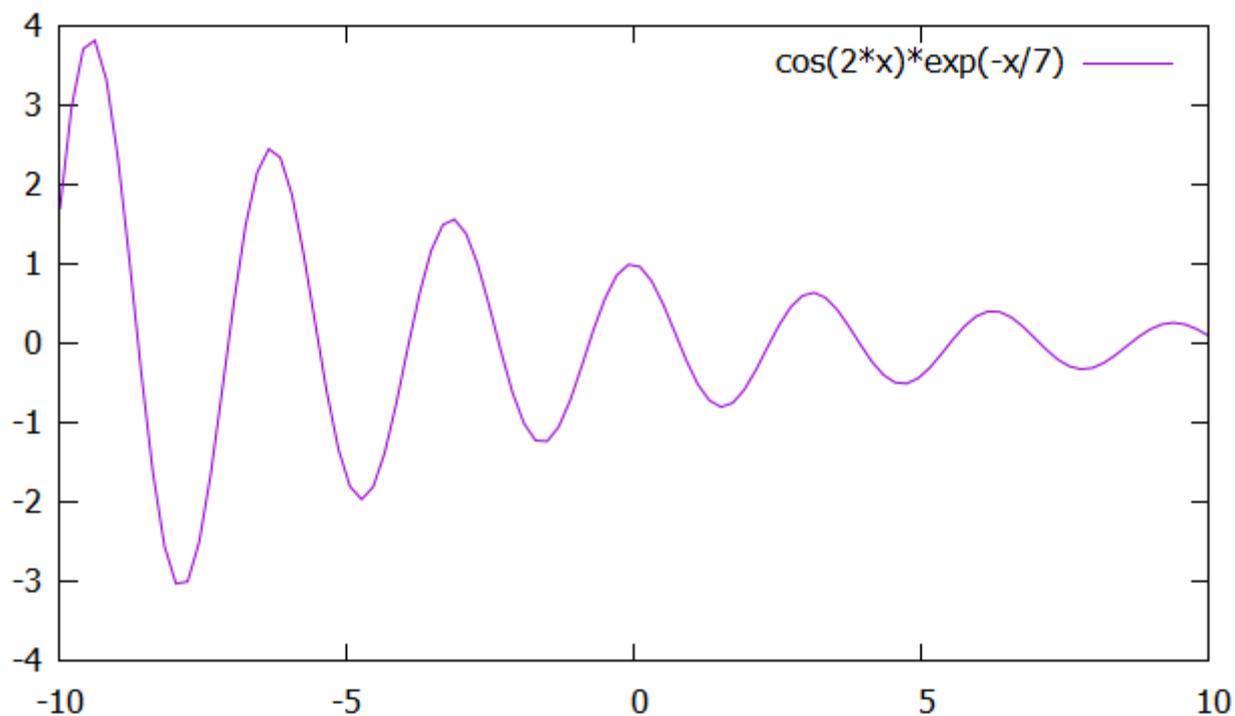
G N U P L O T
Version 5.2 patchlevel 2      last modified 2017-11-15

Copyright (C) 1986-1993, 1998, 2004, 2007-2017
Thomas Williams, Colin Kelley and many others

gnuplot home:      http://www.gnuplot.info
faq, bugs, etc:    type "help FAQ"
immediate help:    type "help"  (plot window: hit 'h')

Terminal type is now 'qt'
gnuplot>
```

- Команда `plot cos(2*x)*exp(-x/7)` отобразит график требуемой функции:

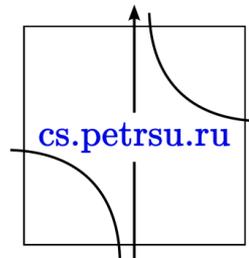


Перенаправление вывода:

- `set terminal postscript`
- `set output mycos.ps`

- `set terminal png`
- `set output mycos.png`

- `set terminal eps`
- `set output mycos.eps`



Диапазоны изменения аргумента

- При построении графика:

```
plot [-3.5:4] [-2:2] cos(2*x)*exp(-x/7)
```

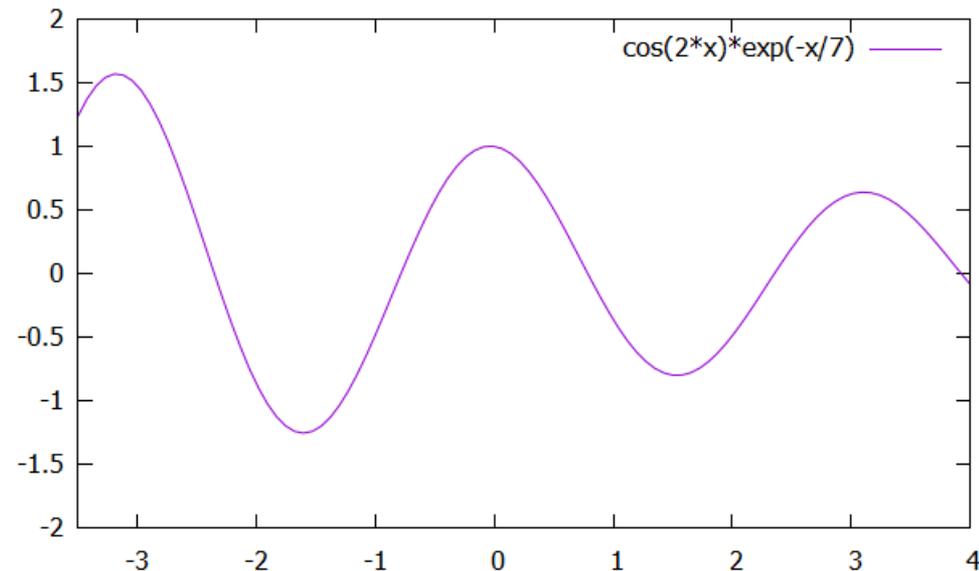
```
plot [] [-2:2] cos(2*x)*exp(-x/7)
```

- Или командами:

```
set xrange [-3.5:4];
```

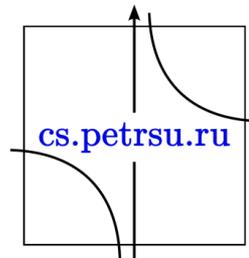
```
set yrange [-2:2];
```

```
plot cos(2*x)*exp(-x/7)
```



Некоторые команды:

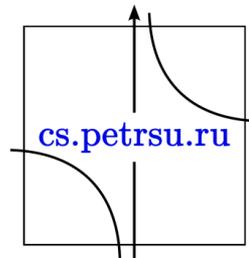
- `set par-p` – установка параметра
- `show xrange` - действующую установку по оси абсцисс
- `show all` - узнать все действующие установки
- `reset` – сбросить все установки по умолчанию
- `help` – вывод справки
- `quit` или `exit` – выход из `gnuplot`



Вывод нескольких графиков

- Осуществляется перечислением в команде `plot` через запятую:

```
plot [-3:3] [-1.2:1.5] sin(x), cos(x), 0.5*sin(2*x)
```

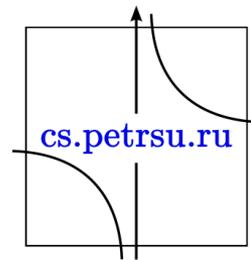


Вывод графика функции, заданной таблично

- Вывод осуществляется указанием файла с входными данными:

```
gnuplot> plot 'gnuexam1.dat' with lines 1
```

```
gnuplot> splot "file.dat"
```



```
GNUPLOT
Version 5.2 patchlevel 8    last modified 2019-12-01
```

```
Copyright (C) 1986-1993, 1998, 2004, 2007-2019
Thomas Williams, Colin Kelley and many others
```

```
gnuplot home:      http://www.gnuplot.info
faq, bugs, etc:   type "help FAQ"
immediate help:   type "help" (plot window: hit 'h')
```

Terminal type is now 'wxt'

Encoding set to 'cp1251'.

```
gnuplot> set title "Построение поверхности в цилиндрических координатах из файла данных."
```

```
gnuplot> set mapping cylindrical
```

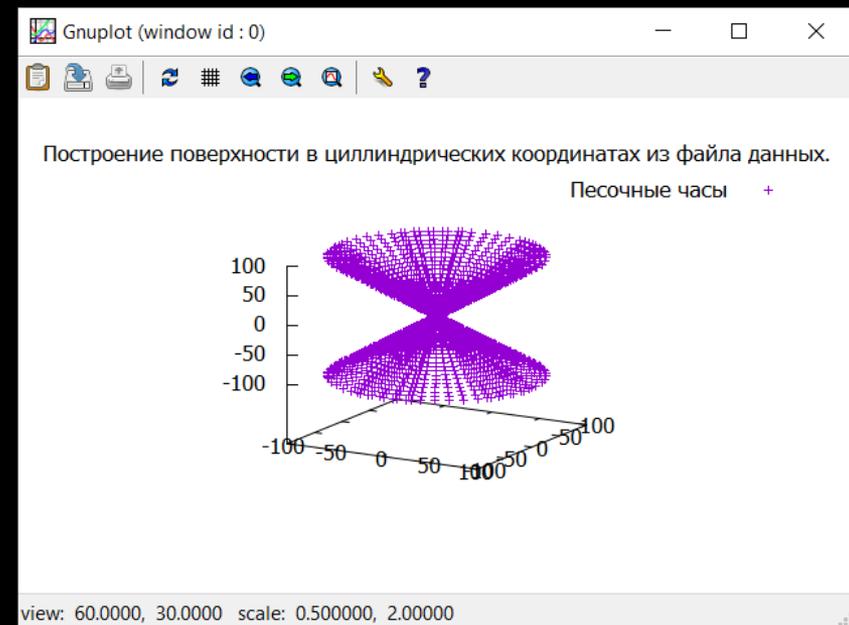
```
gnuplot> set xtics -100, 50, 100
```

```
gnuplot> set ytics -100, 50, 100
```

```
gnuplot> set view 60, 30, 0.5, 2
```

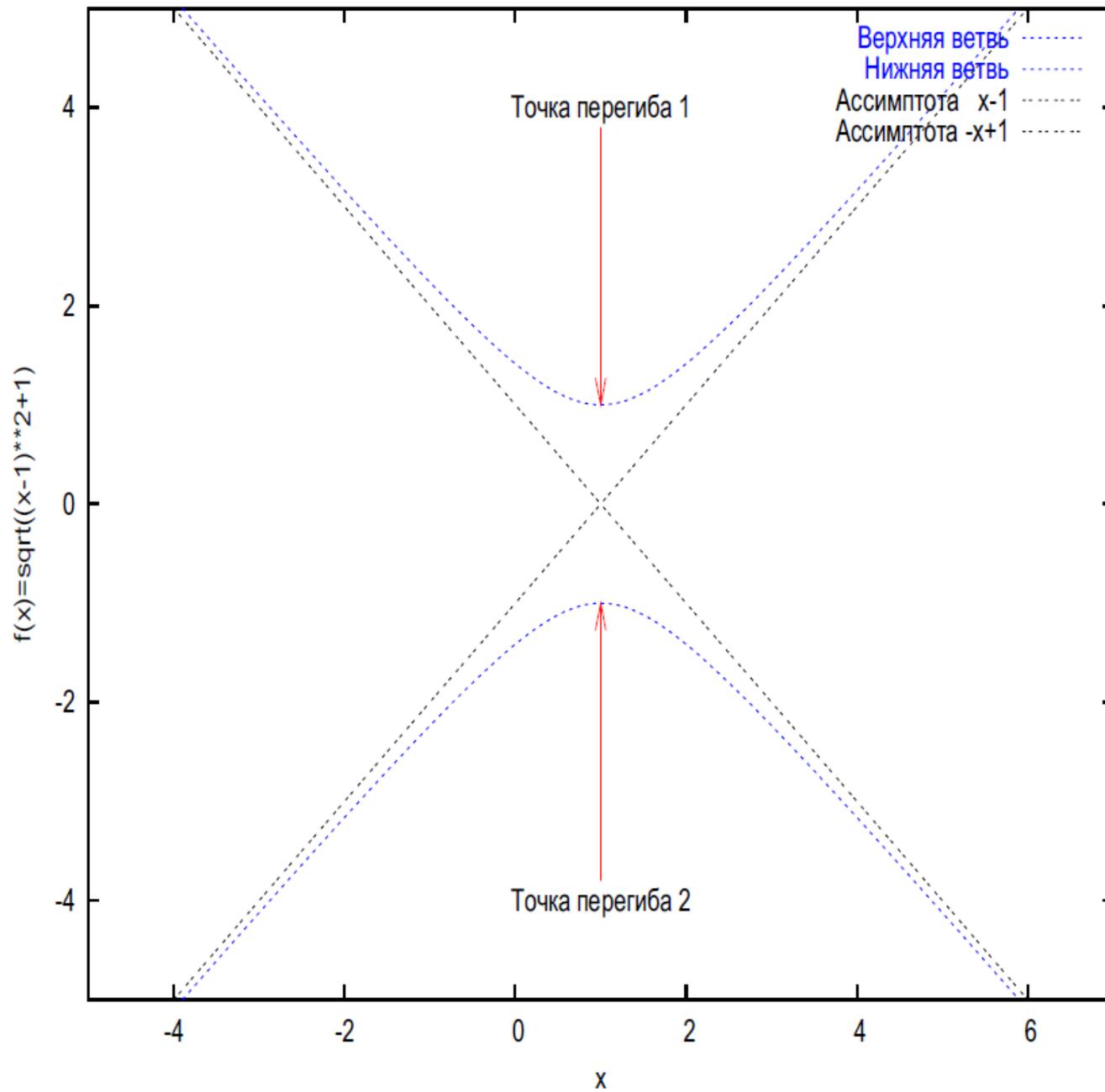
```
gnuplot> splot "cyl.file" title "Песочные часы" w point
```

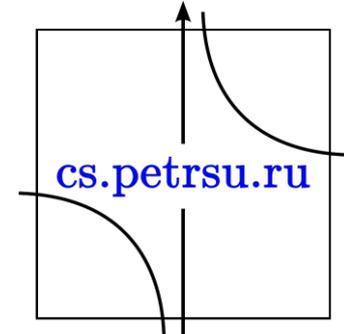
```
gnuplot>
```



```
set terminal postscript eps color "ArialCyrMT" 14
set output "N1.eps"
set xlabel "x"
set ylabel "f(x)=sqrt((x-1)**2+1)"
set label 1 "Точка перегиба 1" at 1, 4 center
set arrow 1 from 1, 3.8 to 1,1
set label 2 "Точка перегиба 2" at 1, -4 center
set arrow 2 from 1, -3.8 to 1,-1
plot [-5:7][-5:5] sqrt((x-1)**2+1) title "Верхняя ветвь"
w l 3, -sqrt((x-1)**2+1) title "Нижняя ветвь" w l 3, x-1
title "Ассимптота x-1" w l 16, -x+1 title "Ассимптота -
x+1" w l 16
```







Визуальное представление математических объектов

Команды и параметры Gnuplot

Глава №2

- `cd` – изменение текущего каталога

- `call`

*call "<input-file>" <parameter-0> <parm-1> ...
<parm-9>*

- `clear`

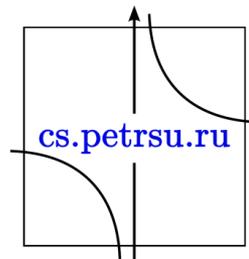
- `exit`

- `fit`

*fit {[xrange] {[yrange]}} <function> '<datafile>'
{datafile-modifiers}
via '<parameter file>' | <var1>{,<var2>,...}*

- `help` – помощь

help plot



- **history**

- **if**

*if (<condition>) <command-line> [; else if (<condition>)
...; else ...]*

- **load**

load "<input-file>"

- **pause**

pause <time> {"<string>"}

*pause mouse {<endcondition>}{, <endcondition>}
{"<string>"}*

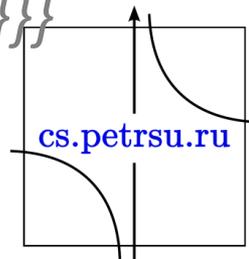
- **plot**

plot {<ranges>}

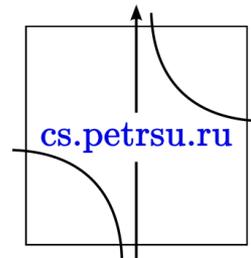
{<function> | {"<datafile>" {datafile-modifiers}}}

{axes <axes>} {<title-spec>} {with <style>}

{, {definitions,} <function> ...}



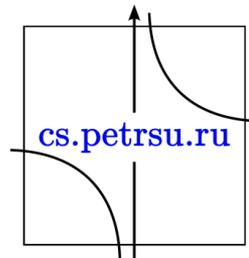
- **print**
print <expression> {, <expression>, ...}
- **pwd** – вывод текущего каталога
- **quit**
- **replot**
plot '-' ; ... ; replot
- **reread**
перечитывает заново gnuplot-файл, в котором находится
- **reset**
убрать все пользовательские изменения
- **save**
save {<option>} '<filename>'



Команда set

set angles {degrees | radians}

set arrow {<tag>} {from <position>} {to | rto <position>}
 { {arrowstyle | as <arrow_style>}
 | { {nohead | head | backhead | heads}
 {size <length>,<angle>{,<backangle>}}
 {filled | empty | nofilled}
 {front | back}
 { {linestyle | ls <line_style>}
 | {linetype | lt <line_type>}
 {linewidth | lw <line_width>} } } }



```
set autoscale {<axes>{ | min | max | fixmin | fixmax | fix }  
| fix | keepfix}
```

```
set parametric
```

```
set polar
```

```
set bars {small | large | fullwidth | <size>}
```

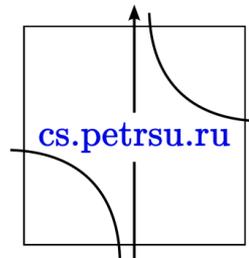
```
set margin
```

```
set bmargin {<margin>}
```

```
set lmargin {<margin>}
```

```
set rmargin {<margin>}
```

```
set tmargin {<margin>}
```

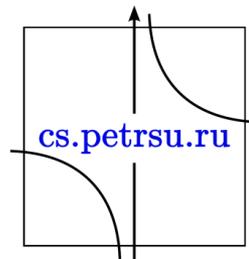


```
set border {<integer>} {front | back} {linewidth | lw <line_width>}  
          {{linestyle | ls <line_style>} | {linetype | lt <line_type>}}
```

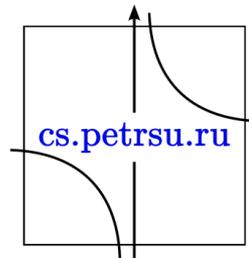
```
set boxwidth {<width>} {absolute | relative}
```

```
set clip <clip-type>
```

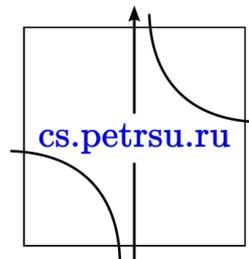
```
set cntrparam { { linear  
                | cubicspline  
                | bspline  
                | points <n>  
                | order <n>  
                | levels { auto {<n>} | <n>  
                        | discrete <z1> {,<z2>{,<z3>...}}  
                        | incremental <start>, <incr> {,<end>}  
                }  
            }  
}
```



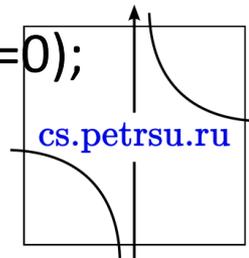
- `set grid` – установка сетки
- `set xlabel "x"`
- `set ylabel " y"`
- `set key center` - легенда
- `set nokey` – убрать легенду

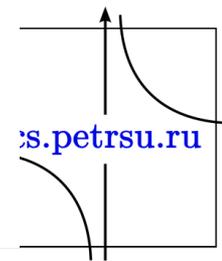
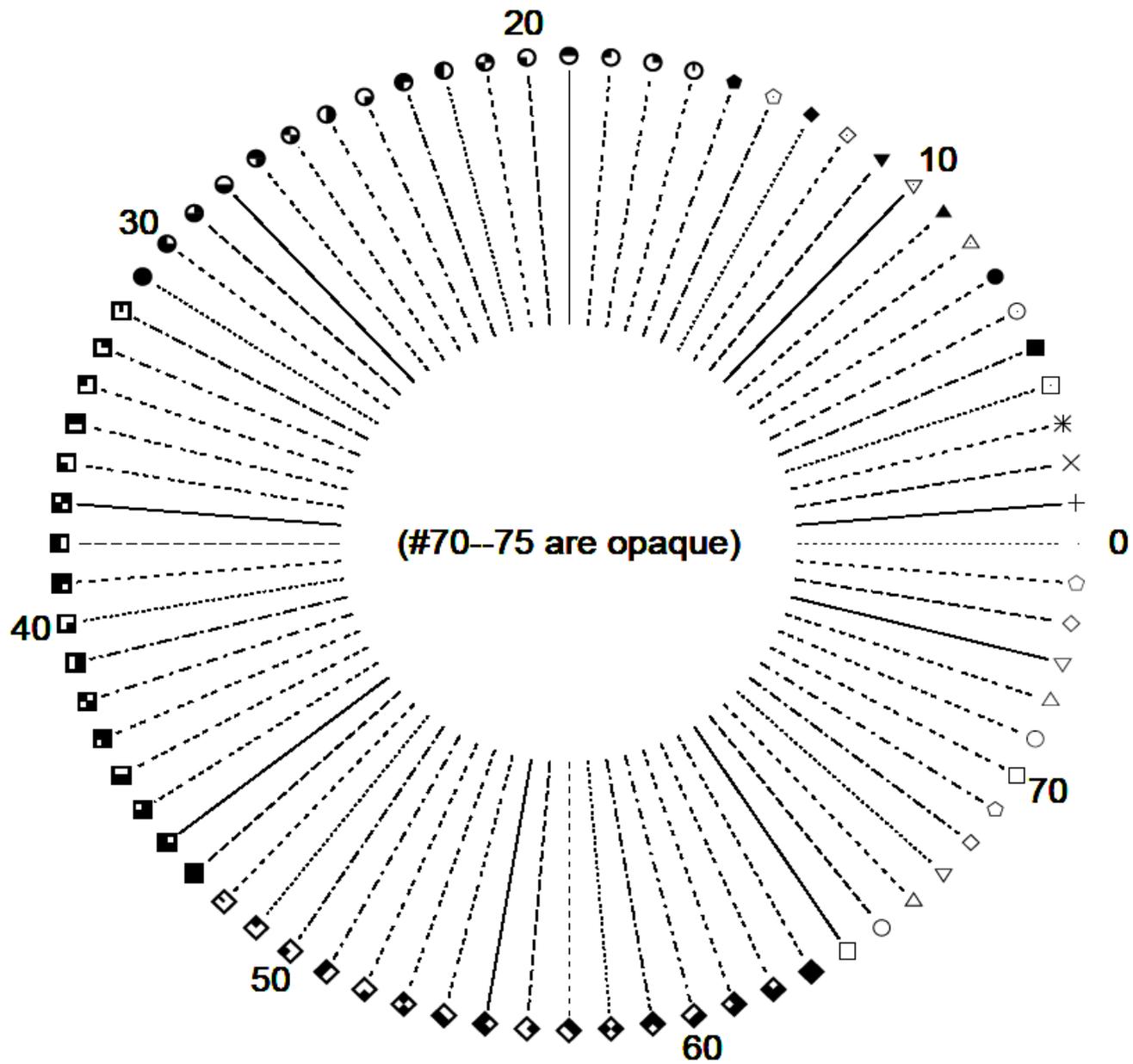


```
plot [1:5][1:120] \  
  for [i = 1:3] "file_" . i . ".dat" \  
  pointsize 1.3 linecolor i+4 \  
  title "file\__" . i . ".dat" \  
  with linespoint
```



- При желании видеть непрерывную кривую достаточно дополнить команду plot опцией with, задающей стиль вычерчивания
 - После ключевого слова with пишется одно из имен, задающих образец линии, выбранной для графика.
 - Запись имени возможна в сокращенной форме (например, достаточно одной буквы w вместо with или l вместо lines).
1. lines означает линию, стиль и толщина которой задаются их номерами после опций lt (linetype) и lw (linewidth) соответственно.
 2. points – дискретный символ (значок), вид и размер которого задаются аналогично после опций pt (pointtype) и ps (pointsize).
 3. linespoints – линия с распределенными по ней значками; возможно совместное применение опций lt, lw, pt и ps.
 4. dots – просто точки;
 5. impulses – отрезки ординат (всплесков) от оси абсцисс ($y=0$);

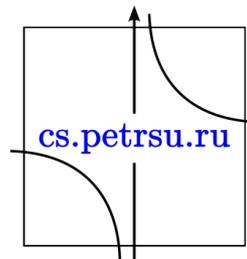
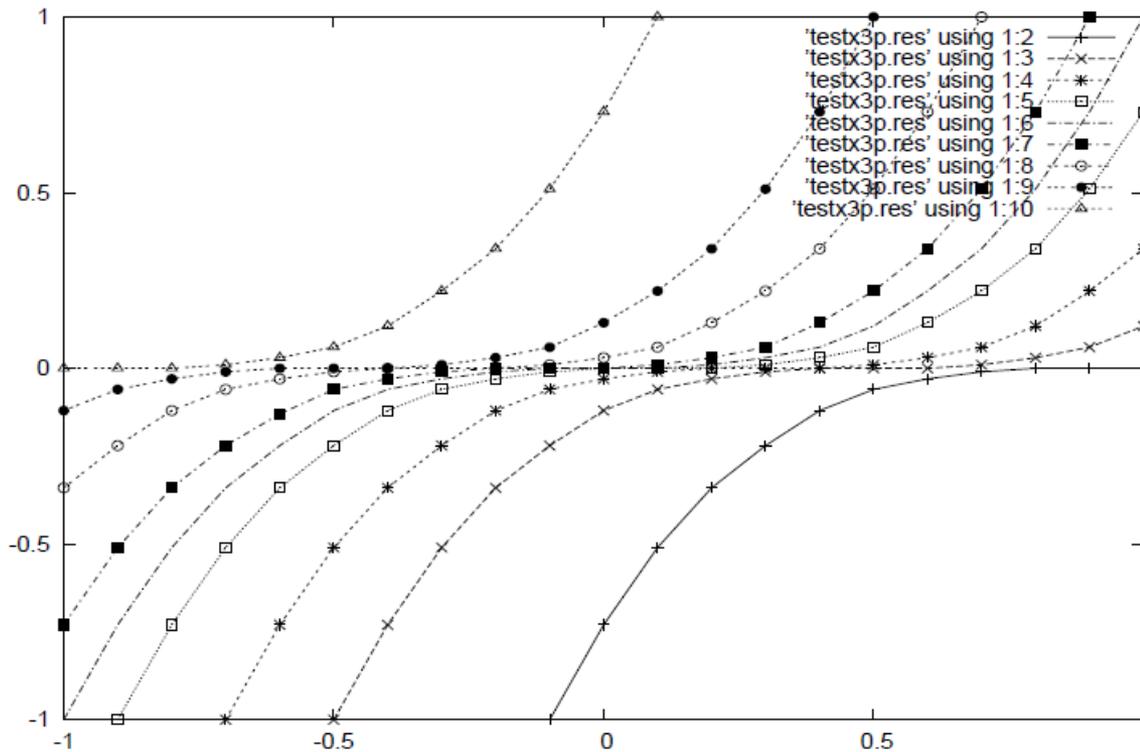




```

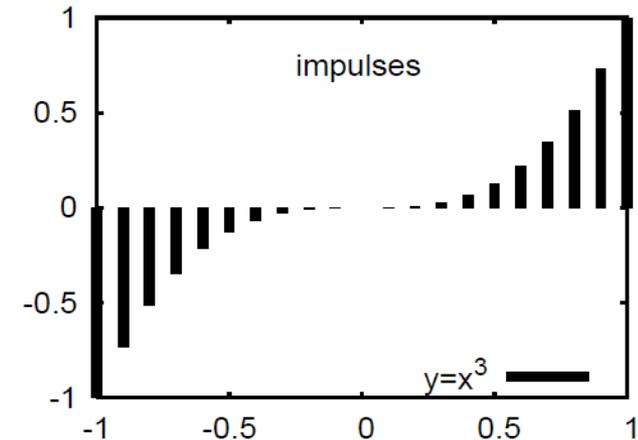
set terminal postscript eps enhanced
set output 'testx3p0.eps'
plot [-1:1] [-1:1]
      'testx3p.res' using 1:2 w lp , \
      'testx3p.res' using 1:3 w lp , 'testx3p.res' using 1:4 w lp , \
      'testx3p.res' using 1:5 w lp , 'testx3p.res' using 1:6 w l  , \
      'testx3p.res' using 1:7 w lp , 'testx3p.res' using 1:8 w lp , \
      'testx3p.res' using 1:9 w lp , 'testx3p.res' using 1:10 w lp

```

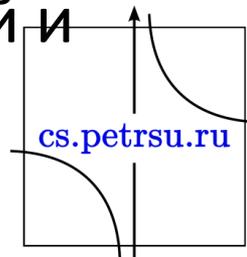


Представление значений функции

- Опция `impulses`:
`set key bot right`
`set border 15 lw 3`
`set label "impulses" at -0.25, 0.75`
`plot "x3.dat" title "y=x^3" w impulses lw 20`



- Опция `impulses` удобна для отображения δ -образных всплесков; наглядно выделяет пропуски во временных рядах, может оказаться полезной и при построении гистограмм.



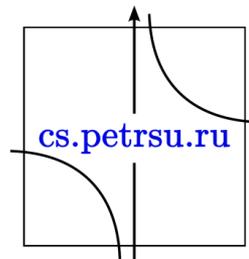
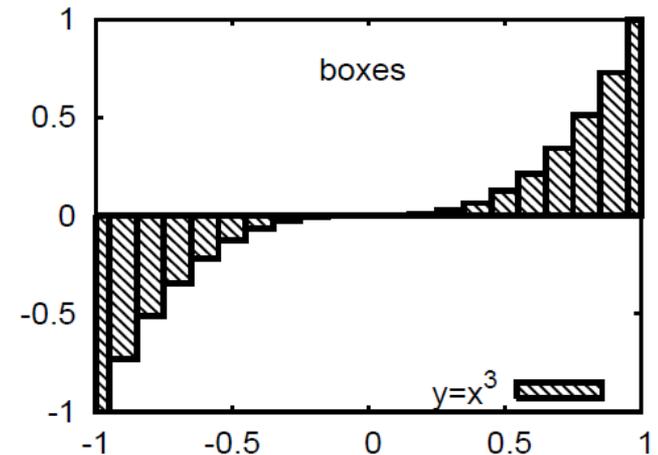
- Опция boxes.

set key bot right;

set border 15 lw 3

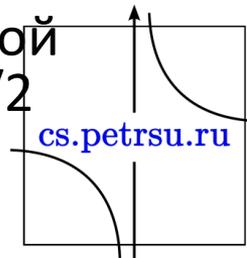
set label "boxes" at -0.25, 0.75

plot 'x3.dat' title "y=x^3" w boxes lw 20



Опции steps, fsteps и histeps.

- Позволяют отобразить график ступенчатой фигурой с постоянным значением функции в пределах одного шага. Различие заключено в очередности и способе черчения горизонтальной и вертикальной компонент ступени:
 - 1. steps чертит сначала горизонтальную часть k-ой ступеньки отрезком прямой $y = y_k$, ограниченной прямыми $x = x_k$ и $x = x_{k+1}$;
 - 2. fsteps чертит сначала вертикальную часть k-ой ступеньки отрезком прямой $x = x_k$, ограниченный прямыми $y = y_k$ и $y = y_{k+1}$;
 - 3. histeps работает в стиле steps, однако вертикальная компонента проводится не в узловой точке, а в срединной точке каждого шага по аргументу, т.е. в точке $(x_k + x_{k+1})/2$

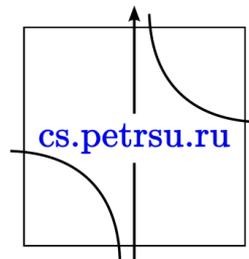


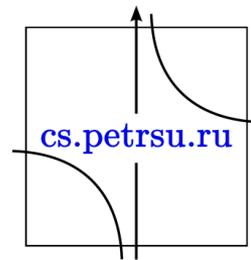
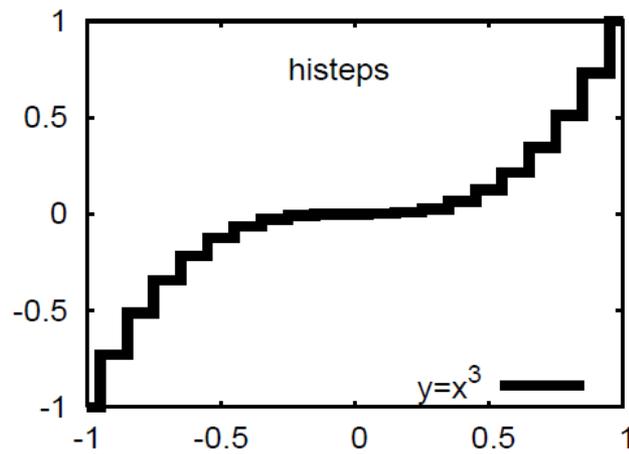
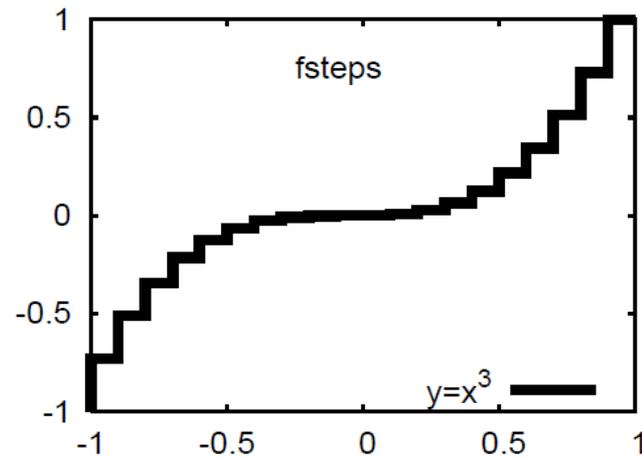
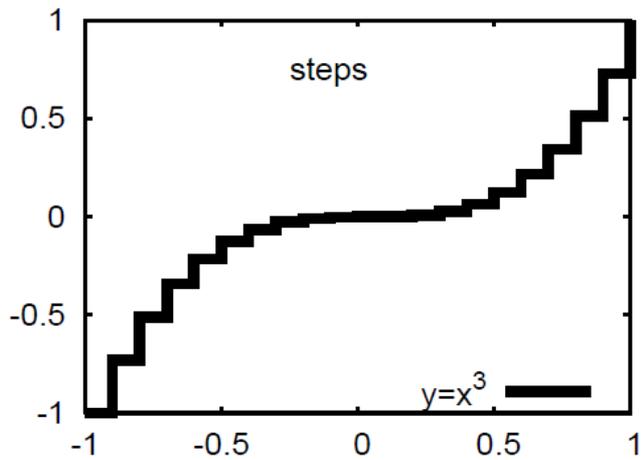
```
set key bot right; # место легенды (внизу, справа)
```

```
set border 15 lw 3 # задаем вид рамки
```

```
set label "steps" at -0.25, 0.75 # текст метки и координаты
```

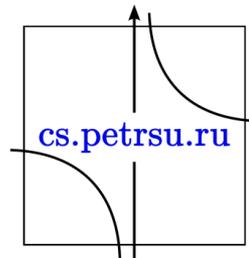
```
plot 'x3.dat' title "y=x^3" w steps lw 20
```





Пример LaTeX-файла

```
\documentclass[a4paper,12pt]{report}
\usepackage[utf8]{inputenc}
\usepackage[english,russian]{babel}
\usepackage{epsfig}
\usepackage{verbatim}
% В текст будут включены объекты Postscript
\usepackage{epsfig, graphicx, euscript}
\tolerance=400
\pagestyle{plain}
\title{Пример включения eps-файла в документ}
\author{О. Ю. Богоявленская}
\date{}
```



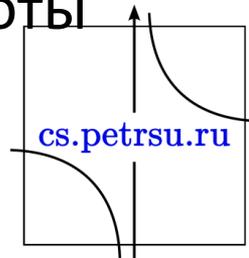
```

\begin{document}
\begin{figure}
\begin{center}
{\includegraphics[width=0.73\textwidth, height=0.27\textheight]
{w-dist-s.eps}}
\quad\mbox{}
\end{center}
\vspace{-7mm}
\caption{Распределение функции частот.}
\label{freq_dist}
\end{figure}

```

На Рис. \ref{freq_dist} приведено распределение частоты событий

```
\end{document}
```



Второй вариант (overleaf.com):

```
\documentclass[a4paper,12pt]{report}
```

```
\usepackage[utf8]{inputenc}
```

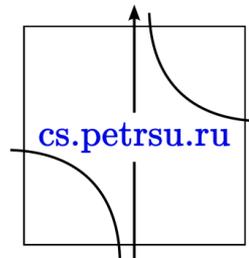
```
\usepackage[english,russian]{babel}
```

```
\usepackage{epsfig}
```

```
\usepackage{verbatim}
```

```
\usepackage{gnuplottex}
```

```
\begin{document}
```



```
\section{Построение графиков Gnuplot в документе \LaTeX}
```

```
\begin{figure}[h]
```

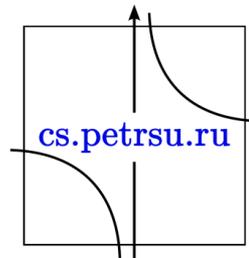
```
\centering
```

```
\begin{gnuplot}
```

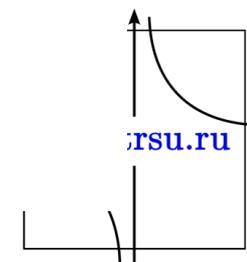
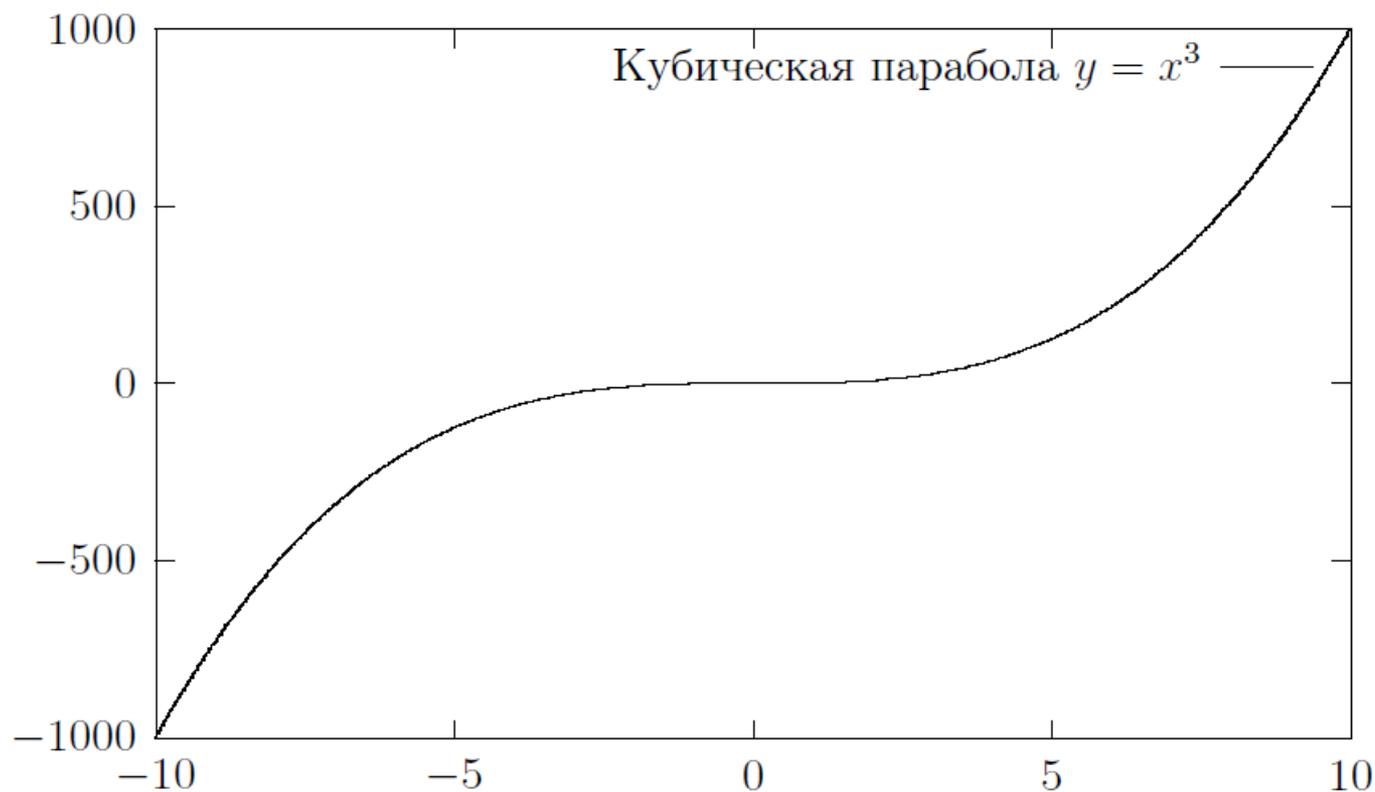
```
plot x**3 title 'Кубическая парабола  $y = x^3$ '
```

```
\end{gnuplot}
```

```
\end{figure}
```



0.1 Построение графиков Gnuplot в документе ЛАТ_EX



- Примеры от разработчика:
http://gnuplot.sourceforge.net/demo_4.6/
- Ещё примеров: <http://www.gnuplotting.org/>
- <http://gnuplot-tricks.blogspot.com/>

