
Open Source Campus Agreement

Modul Pelatihan

GNU PLOT UNTUK ORANG LUGU

oleh
ABe Susanto dan I Wayan Warmada
Tim PANDU

Editor:
I Made Wiryana

Hak cipta buku ini tetap pada penulis. Tetapi buku ini bebas untuk diperbanyak, dikutip baik sebagian atau seluruhnya ataupun disebar luaskan dalam bentuk elektronik ataupun non-elektronik. Baik untuk tujuan komersial maupun non komersial. Selama penyebutan nama asli pengarang, penerbit, pemberi sponsor serta proyek Open Source Campus Agreement (OSCA) tetap dilakukan.

2001

Gnuplot untuk orang lugu

ABe Susanto <susanto@zfn.uni-bremen.de>
I Wayan Warmada <warmada@nakula.rvs.uni-bielefeld.de>

Editor: I Made Wiryana <mwiryana@rvs.uni-bielefeld.de>

2001

Daftar Isi

Kata Pengantar	iv
Tentang penulis	v
Pernyataan	vi
1 Pengenalan gnuplot	1
1.1 Pendahuluan	1
1.2 Menginstal gnuplot	2
2 Mengenal perintah sederhana gnuplot	4
2.1 Format data pada gnuplot	4
2.2 Perintah sederhana gnuplot	5
2.3 Menyimpan tampilan grafik dalam format lain	7
2.4 Latihan	8
3 Membuat grafik multiplot dan errorbar	9
3.1 Multiplot dalam satu grafik	9
3.2 Multiplot dalam satu halaman kertas	10
3.3 Membuat grafik errorbars	12
3.4 Latihan	14
4 Ekstrapolasi	16
4.1 Melakukan fitting data	16
4.2 Latihan	17
5 Data multivariat	18
5.1 Grafik 3 dimensi	18
5.2 Menggambar kontur	20
5.2.1 Data teratur	20
5.2.2 Data tidak teratur (acak)	21
5.3 Menggambar diagram segitiga	22
5.4 Latihan	24
6 Mengatur tampilan grafik	25
6.1 Seting terminal PostScript	25
6.2 Menambahkan teks dan panah pada grafik	26
6.3 Menset ukuran grafik	27
6.4 Mengatur lebar margin	28
6.5 Latihan	28

7 Menggunakan gnuplot untuk L^AT_EX	29
Lampiran A. Alamat beberapa situs grafik	32
Lampiran B. Data tambahan	33

Daftar Gambar

2.1	Contoh hasil pemplotan dengan tipe <code>boxes</code>	5
2.2	Contoh hasil pemplotan dengan tipe <code>lines</code>	6
2.3	Contoh hasil pemplotan dengan tipe <code>linespoints</code>	6
2.4	Contoh hasil pemplotan dengan tipe <code>points</code>	6
3.1	Tampilan hasil pemplotan multigrafik	10
3.2	Susunan empat buah grafik pada perintah <code>multiplot</code>	11
3.3	Tampilan hasil pemplotan multigrafik dalam satu halaman kertas	12
3.4	Tampilan grafik tipe <code>errorbars</code>	13
3.5	Tampilan grafik hasil <code>multiplot</code> tipe <code>errorbars</code>	14
4.1	Contoh ekstrapolasi linear dengan fungsi $f(x) = y_0 + m \cdot x$	17
5.1	Contoh hasil pemplotan data 3 dimensi	19
5.2	Contoh hasil perubahan sudut pandangan	19
5.3	Gambar kontur yang diplot dari data teratur tanpa seting <code>cntrparam</code>	21
5.4	Gambar kontur yang dihasilkan dari data acak dengan seting <code>cntrparam cubicspline</code>	22
5.5	Prinsip pemplotan dan pembacaan diagram segitiga	23
5.6	Hasil pemplotan skrip diagram segitiga	24
6.1	Pengesetan terminal <code>postscript</code>	26
6.2	Gambar peta Indonesia dan negara tetangga	27
7.1	Gambar yang dihasilkan dari dari file <code>L^AT_EX</code> keluaran skrip di atas	30

Kata Pengantar

Saat ini dalam dunia komputer di negara-negara seperti Amerika, Australia, India, Inggris, Jepang, Jerman, negara-negara Skandinavia, dan sebagainya telah berkembang dan sudah banyak orang menggunakan sistem operasi berbasis Linux, yaitu suatu sistem operasi yang bersifat 'open source'. Hal ini berarti sistem operasi tersebut bisa disalin dan diinstal dari sebuah komputer ke komputer lainnya tanpa ada pembayaran lisensi, selain itu bagi programmer yang gemar 'mengutak-ngatik source code' nya akan mendapatkan tempat yang cocok, karena hal itu bukanlah dilarang justru sebaliknya malah dianjurkan sehingga menghasilkan suatu sistem operasi yang lebih baik.

Di sisi lain, penulis merasa sudah beberapa tahun mengenal linux sebagai hobi atau penunjang kegiatan utama yaitu menimba ilmu di negeri yang serba tehnologi ini, merasa belum pernah menulis buku linux yang cukup serius. Walaupun tulisan-tulisan lepas penulis tentang Linux ada beberapa yang telah di terbitkan di internet dengan alamat URL adalah <http://ngelmu.dhs.org/> atau <http://pandu.dhs.org/>. Oleh karena itu sebagai rasa kecintaan kami pada Linux dan untuk menambah variasi pustaka tentang Linux yang berbahasa Indonesia, maka penulis merasa terpanggil dan memberanikan diri menguraikan sepintas tentang dunia grafik dengan software `gnuplot`, seperti dalam buku ini.

Buku ini dibagi menjadi beberapa bagian yang terdiri dari pengenalan `gnuplot` mencakup pengantar dan instalasi `gnuplot`; mengenal perintah `gnuplot` yang membahas tentang format data, contoh perintah sederhana, menyimpan tampilan `gnuplot`; grafik multiplot, ekstrapolasi, data multivariat. Bab berikutnya adalah membuat grafik dengan standar deviasi dan beberapa contoh aplikasi `gnuplot` pada bidang lainnya. Kemudian semua uraian dalam buku ini lebih cocok digunakan bagi pemula yang belum pernah mengenal `gnuplot` sama sekali. Hal ini disebabkan contoh dalam penggunaannya diberikan secara mudah untuk dicerna dan sangat sederhana.

Pada kesempatan yang baik ini, penulis juga menyampaikan terima kasih kepada teman-teman seangkatan penerima beasiswa DAAD yang telah mendorong penulis memberikan semangat untuk menyelesaikan buku ini, baik berupa kritikan atau masukan yang membangun. Selain itu penulis juga akan berterima kasih, apabila mendapatkan kritikan dan saran bersifat membangun tentang materi yang ditulis dalam buku ini guna perbaikan penulisan di kemudian hari. Saran dan kritikan dari pembaca dapat dikirimkan melalui e-mail langsung ke penulis

St. Pauli, Desember 2000

Penulis

- **ABe Susanto** <susanto@zfn.uni-bremen.de>
- **I Wayan Warmada** <warmada@nakula.rvs.uni-bielefeld.de>

Tentang penulis



Ir. I Wayan Warmada, staf Fakultas Teknik, Universitas Gadjah Mada. Merampungkan S1 bidang geologi di Universitas Gadjah Mada. Dengan beasiswa dari DAAD kini sedang melanjutkan studi doctoral bidang Geologi Ekonomi di Technical University of Clausthal di bawah bimbingan Prof.Dr. Bernd Lehmann. Penelitiannya mendalami proses pembentukan Emas Epithermal. Pernah mengajar mata kuliah Komputer untuk Geologi dan Gambar Teknik Geologi yang menggunakan komputer sebagai alat bantu utama. Sangat tertarik untuk mencoba hampir semua program grafis baik yang jalan di Linux, Solaris maupun Windows (hanya untuk membandingkan efisiensi kerjanya). Pengalaman berkomputer banyak diperoleh ketika melakukan "Summer Job" di PT Caltex Pacific Indonesia (1994). Mengasuh forum Corat Coret Cantik suatu bahasan tentang penggunaan perangkat lunak grafik alternatif di situs TIM PANDU.



Drs. ABe Susanto, MSc, menekuni bidang kajian RUMPUT LAUT selama menyelesaikan studi S1 di Unsoed Purwokerto dan S2 di Universitas Ryukyus, Jepang. Staf muda dari Fakultas Perikanan dan Ilmu Kelautan UNDIP ini sekarang sedang "ngelmu" tentang rumput laut lagi di Fak. Biologi/Kimia Universitas Bremen, Jerman di bawah bimbingan Prof.Dr.Gunter O. Kirst dan Prof.Dr. Christian Wiencke dari AWI-Bremerhaven dengan beasiswa dari DAAD. Mata kuliah Pengenalan Komputer pernah diberikan di Jurusan Ilmu Kelautan-FPIK-UNDIP dan SCUBA-Diving adalah kegiatan waktu senggangnya. Mengenal LINUX sejak tiba di Bremen (1997) dan tertarik dengan program Linux yang dapat diaplikasikan di dunia penelitian kelautan. MikroZNet adalah proyek iseng pertama kalinya dengan I Made Wiryana yang mencoba mengkombinasikan LINUX dengan Mikroskop. Keterlibatannya di LINUX mendorong dia bersama-sama dengan rekan lainnya membentuk TIM PANDU



I Made Wiryana SSi SKom MSc menamatkan S1 di jurusan Fisika FMIPA Universitas Indonesia pada bidang instrumentasi dan fisika terapan. Dengan beasiswa dari STMIK Gunadarma juga menamatkan S1 Teknik Informatika di STMIK Gunadarma. Melanjutkan studi S2 di Computer Science Department Edith Cowan University - Perth dengan beasiswa ADCSS dan STMIK Gunadarma pada bidang fuzzy system dan artificial neural network untuk pengolahan suara. Menangani perancangan dan implementasi jaringan Internet di Universitas Gunadarma dengan memanfaatkan sistem Open Source sebagai basisnya. Pernah mewakili IPKIN dalam kelompok kerja Standardisasi Profesi TI untuk Asia Pasifik (SEARCC). Saat ini dengan beasiswa dari DAAD melanjutkan studi doctoral di RVS Arbeitsgruppe Universität Bielefeld Jerman di bawah bimbingan Prof. Peter B Ladkin PhD. Aktif menjadi koordinator beberapa proyek penterjemahan program Open Source seperti KDE, SuSE, Abiword dan juga sebagai advisor pada Trustix Merdeka, distribusi Linux Indonesia yang pertama. Terkadang menyumbangkan tulisannya sebagai kolumnis pada media online DETIK.COM dan SATUNET. Juga kontributor pada KOMPUTEK, MIKRODATA, ELEKTRO dan INFOLINUX. Kontribusi ke komunitas Open Source sering dilakukan bersama-sama kelompok Tim PANDU. Star pengajar tetap Universitas Gunadarma.

Pernyataan

Penulisan buku dilakukan selama penulis melanjutkan studi di Jerman atas beasiswa dari DAAD. Secara tidak langsung penulisan buku ini disponsori oleh:

- **Universtas Gadjah Mada**

Bulaksumur, Yogyakarta

URL : <http://www.ugm.ac.id>

- **Universitas Diponegoro**

URL : <http://www.undip.ac.id>

- **Deutscher Akademischer Austauschdienst (DAAD)**

Kennedyalle 50

D-53175 Bonn - Jerman.

URL : <http://www.daad.de>

Email : daadjak@rad.net.id

Proses pengeditan dan pemformatan dilakukan editor secara tidak langsung juga disponsori oleh : Deutscher Akademischer Austauschdienst (DAAD) dan

- **Universitas Gunadarma**

Jl Margonda Raya No 100. Depok - Jakarta

URL : <http://www.gunadarma.ac.id>

Beberapa merk dagang yang disebutkan pada buku ini merupakan merk dagang terdaftar dari perusahaan tersebut, kecuali bila disebutkan lain.

Bab 1

Pengenalan gnuplot

1.1 Pendahuluan

Gnuplot adalah suatu program grafik interaktif yang berbasis CLI (*command-line interface*). Program ini sangat mudah digunakan dan sebenarnya hanya memiliki 2 perintah untuk membuat suatu gambar plot, yaitu perintah : `plot` dan `splot`. Program ini dapat menghasilkan banyak macam grafik dengan banyak pilihan untuk mengubahnya dan dapat disimpan ke dalam bermacam-macam divais grafik (terminal grafik, printer atau ploter).

Kemudahan dalam penggunaan dan kesederhaan perintah-perintahnya tidaklah mengurangi kualitas kemampuan program ini, oleh karena itu program gnuplot mempunyai beberapa macam kemampuan, yaitu sebagai berikut:

- Memplot deretan data univariat (misalnya time series)
- Memplot sederhana suatu fungsi yang didefinisikan pengguna baik pada sistem koordinat Cartesius maupun polar
- Scatter plot data bivariat dengan pilihan errorbar
- Grafik batang
- Memplot permukaan 3-dimensi suatu fungsi seperti $z = f(x,y)$ dengan beberapa pilihan
- Scatter plot 3-dimensi data trivariat
- Memplot 2- dan 3-dimensi suatu fungsi parametrik
- Memplot data secara langsung dari tabel yang dihasilkan oleh aplikasi yang lain
- Memplot berulang pada suatu variasi devais grafik yang lain.

Di samping program grafik berbasis CLI, terdapat juga program yang berbasis GUI (*Graphical User Interface*) yang banyak disenangi oleh pengguna, misalnya Xmgrace, GUPPI, Scigraphica, Xgfe dan Xgobi. Semua program ini dapat dijalankan pada lingkungan Linux. Xgfe (*X-gnuplot front-end*) merupakan front-end dari gnuplot pada terminal X window

Barangkali timbul suatu pertanyaan, “*mengapa memilih program gnuplot, sedangkan di sisi lain banyak tersedia program grafik berbasis GUI secara bebas?*” Salah satu alasan mengapa memilih program berbasis CLI, misalnya program ini dapat dijalankan lebih cepat dibandingkan program berbasis GUI karena program ini tidak membutuhkan terminal grafik yang bagus. Hanya dengan memakai komputer “butut” pun masih bisa membuat grafik dengan kualitas cetak yang sangat baik. Di samping itu kadang data mentah yang diperoleh dari suatu pengukuran memerlukan suatu proses awal (*preprocessing*) sebelum diplot ke dalam grafik, misalnya memberikan koreksi tertentu. Minimal kita akan bekerja beberapa kali untuk menghasilkan grafik dari data mentah seperti ini. Sebagai contoh, pertama melakukan perhitungan koreksi dan kedua melakukan pemplotan. Memang tidak terlalu sulit melakukan hal seperti ini pada program berbasis GUI karena sudah disediakan spreadsheet untuk melakukan proses awalnya.

Dalam program gnuplot yang berbasis CLI kita tidak perlu melakukan perhitungan seperti itu, karena program ini dapat melakukan perhitungan bersama-sama dengan pemplotan. Dalam perhitungan yang kompleks, bila

spreadsheet tidak dapat melakukannya (atau cukup rumit), `gnuplot` dapat digabung dengan skrip program tertentu, misalnya `perl`, `python`, dan lain-lain. Hal ini yang membuat program ini jauh lebih cepat dalam membuat grafik dibandingkan dengan program berbasis GUI. Alasan ini lah yang membuat `gnuplot` dapat digunakan juga untuk membuat grafik suatu web, misalnya untuk menampilkan grafik analisis web secara on-line.

Penulis sendiri memanfaatkan `gnuplot` untuk menunjang menyelesaikan tugas akhir terutama untuk menampilkan data dalam bentuk grafik. Alasan penulis menggunakan program `gnuplot` adalah karena program ini dapat diperoleh secara gratis dan mempunyai kemampuan untuk menampilkan gambar sesuai yang kita inginkan. Walaupun dalam dunia grafis dikenal juga berbagai macam program grafik populer yang lain, akan tetapi sebagian besar pemakai harus mengeluarkan sejumlah uang yang cukup besar untuk membeli program tersebut. Sebagai contoh saja, program `Xact` untuk versi 'up date', paling sedikit seseorang harus mengeluarkan Rp. 500.000,-(saat ini 1 DM setara dengan Rp. 4.100,-). Program `Xact` sendiri dibuat oleh SciLab GmbH di Hamburg, Jerman.

1.2 Menginstal gnuplot

Ada beberapa cara untuk menginstal program `gnuplot` di lingkungan linux maupun windows. Cara yang paling mudah adalah menginstal `gnuplot` dari file biner (`*.rpm`, `*.deb`, `*.zip`, `*.exe`) atau menginstal dari paket yang terdapat pada distributor seperti SuSE, EasyLinux, RedHat, Debian, Mandrake, dan lain-lain. Biasanya setiap distributor memberikan cara tertentu untuk menginstal program yang ada pada paket program yang dipaketkannya. Lihat panduan instalasi Linux pada tiap-tiap distributor yang digunakan.

Cara yang lain adalah menginstal dari source code `gnuplot` yang dapat didownload melalui situs resmi `gnuplot` (lihat lampiran A). Langkah-langkah pre-instalasi `gnuplot` dari source code adalah sebagai berikut (khusus Linux/Unix):

- Download file `gnuplot-3.7.1.tar.gz` dari salah satu server `gnuplot`. Jika ingin menginstal library GIF (untuk terminal GIF), download file `gd1.4.tar.gz`. Jangan lupa download juga file-file dokumentasinya, misalnya file `gnuplot.pdf.gz`, `gp-ps.zip`, dan lain-lain.
- Ekstrak file yang terkompres tar tersebut dengan perintah `tar -xvzf gnuplot-3.7.1.tar.gz` pada direktori `/usr/src/packages` (hanya untuk memudahkan pencarian). Jangan lupa login sebagai `root` atau `superuser` sebelum memulai instalasi. Proses ekstraksi ini akan membuat direktori `gnuplot-3.7.1` secara otomatis, yang berisi file-file source code `gnuplot`.
- Langkah selanjutnya adalah instalasi.

Pertama, ekstrak dan instal file `gd1.4.tar.gz`, perintahnya sama dengan perintah pada instalasi `gnuplot` yang akan diterangkan berikut ini.

Lokasi default file-file pada program `gnuplot`:

```

PREFIX                /usr/local
gnuplot                PREFIX/bin
gnuplot_x11            PREFIX/bin
gnuplot.1              PREFIX/man/man1
gnuplot.gih            PREFIX/share

```

Perintah instalasi adalah sebagai berikut (`$` artinya direktori tempat source code disimpan, misalnya `root@hiranya: /usr/src/packages/gnuplot-3.7.1 >`):

```

$ ./configure
$ make
$ make install

```

Jika `gcc` terinstal, akan digunakan sebagai default. Jenis kompiler yang lain dapat juga digunakan melalui variabel `CC`:

(Bourne shell atau `bsh`, atau `bash`)

```

$ CC=c89 ./configure

```

(C shell atau csh)

```
$ setenv CC c89
$ ./configure
```

File demo gnuplot tidak diinstal secara default. Jika diinginkan, file-file ini harus disalin secara manual ke suatu direktori yang dipilih, misalnya `/usr/local/share/docs/gnuplot/demo`.

Menginstal gnuplot di lingkungan windows dari source code lebih sulit dibandingkan pada lingkungan Linux atau Unix, karena dibutuhkan suatu kompiler yang harus diinstal terlebih dahulu. Program ini harus dibeli di luar basis sistem operasi MS Windows/DOS. Beberapa kompiler yang bisa dipakai untuk menginstal gnuplot di lingkungan MS Windows, meliputi Microsoft C 7.0, Borland C++, Visual C++, MinGW32 (membutuhkan Microsoft SDK), dan Cygwin (membutuhkan Microsoft SDK). Perintahnya tidak akan dijelaskan di sini, karena pada server gnuplot telah tersedia file biner terkompres program ini. File-file tersebut adalah `gp37w16.zip` (untuk MS Windows 3.1), `gp37w32.exe` dan `gp371w32.zip` (untuk MS Windows 9x/nt). Untuk menginstal file ini dapat dilakukan dengan langkah-langkah sebagai berikut:

- Ekstrak file `gp37w16.zip`, `gp371w32.zip`, atau `gp37w32.exe` (*autoextracted*) pada direktori Programme (hanya untuk pengelompokan supaya lebih mudah dalam pencarian). Gnuplot akan membuat direktori baru, `gp37w16` atau `gp37w32` atau `gp371w32` yang berisi file-file: `wgnuplot.exe` atau `wgnuplot32.exe`, `wgnuplot.dll`, `wgnuplot.hlp`, dan `wgnuplot.mnu`.
- Untuk menjalankan gnuplot ini, panggil file `wgnuplot.exe` atau `wgnuplot32.exe` dari perintah `run`.
- File gnuplot terkompres untuk windows 16 bit (`gp37w16.zip`) tidak menyertakan file-file demo. Jika file ini diinginkan dapat di ekstrak dari source code lalu dikopi.

Jika instalasi dilakukan dengan benar dan gnuplot dijalankan, maka tampilan pembuka gnuplot dapat dilihat seperti berikut ini:

```
$ gnuplot

G N U P L O T
Linux version 3.7
patchlevel 1
last modified Fri Oct 22 18:00:00 BST 1999

Copyright(C) 1986 - 1993, 1998, 1999
Thomas Williams, Colin Kelley and many others

Type 'help' to access the on-line reference manual
The gnuplot FAQ is available from
<http://www.ucc.ie/gnuplot/gnuplot-faq.html>

Send comments and requests for help to <info-gnuplot@dartmouth.edu>
Send bugs, suggestions and mods to <bug-gnuplot@dartmouth.edu>

Terminal type set to 'x11'
gnuplot>
```

Bab 2

Mengenal perintah sederhana gnuplot

2.1 Format data pada gnuplot

Gnuplot dapat menghasilkan bermacam-macam grafik dari data yang terdistribusi. Kaidah penting dari organisasi data adalah mekanisme gnuplot yang membaca data tiap baris. Contoh data yang paling sederhana adalah data seri yang terdiri dari koordinat- x dan koordinat- y . Perintah `plot '<data-filename>'` akan memplot nilai data seri yang terdapat pada file tersebut.

File yang mengandung data harus disimpan dalam bentuk ASCII dengan nama misal `fileku.dat`. Hal yang penting dalam membuat file ASCII ini adalah data-data berupa angka yang disusun berdasarkan kolom dan di antara kolom satu dengan lainnya dipisahkan satu atau lebih spasi. Data yang dapat dibaca oleh program ini adalah teks atau ASCII dalam bentuk baris dan kolom yang tidak terbatas (tergantung kemampuan komputer untuk mengolahnya). Format file data adalah sebagai berikut:

```
# contoh file ---> disimpan dengan nama bebas
# tanda '#' tidak dibaca oleh gnuplot, seperti 'rem' pada program basic
#
X1,1 Y1,2 Y1,3 Y1,4 ... Y1,N
X2,1 Y2,2 Y2,3 Y2,4 ... Y2,N
X3,1 Y3,2 Y3,3 Y3,4 ... Y3,N
X4,1 Y4,2 Y4,3 Y4,4 ... Y4,N
.
.
.
XN,1 YN,2 YN,3 YN,4 ... YN,N
```

Contoh data:

```
# contoh file fileku.dat
#
0 37.76
1 31.30
2 29.33
4 26.65
8 21.65
16 14.68
32 7.45
42 5.48
```

Hampir semua jenis program spreadsheet seperti Microsoft Excel, StarCalc, Lotus 1-2-3, Quatro, Gnumeric atau spreadsheet yang disediakan oleh program grafik yang lain misalnya Scigraphica, Grapher, Origin, Sigmaplot, SPSS, dsb. dapat dipakai untuk menghasilkan data dalam format ASCII. Caranya yaitu dengan memakai perintah `save as` pada menu `File` ke dalam format teks (`*.txt`) dan dipilih karakter pemisah antar kolomnya. Dalam program spreadsheet biasanya ada 3 pilihan untuk memisahkan data berdasarkan kolom dalam

format ini, yaitu spasi, tab dan koma. Gnuplot hanya bisa membaca pemisah kolom berupa spasi dan/atau tab.

2.2 Perintah sederhana gnuplot

Secara default gnuplot akan menseset terminal keluaran ke terminal `x11` atau windows (monitor). Perintah utama untuk menampilkan grafik pada program ini adalah `plot` dan `splot`. Perintah `plot` digunakan untuk memplot grafik dua dimensi, sedangkan perintah `splot` untuk memplot grafik tiga dimensi. Mari kita lihat contoh perintah berikut ini.

```
gnuplot> plot [-pi:pi*2] sin(x)
gnuplot> plot f(x) = sin(x*a), a = .2, f(x), a = .4, f(x)
gnuplot> plot [t=1:10] [-pi:pi*2] tan(t)
```

Selain untuk memplot fungsi matematis di atas, dengan perintah sederhana dapat diplot suatu data numerik. Sebagai contoh, membuat diagram balok dari sebuah penelitian terhadap pertumbuhan rumput laut *Gracilaria sp.*, *Sargassum sp.* dan *Caulerpa sp.* File data hasil penelitian adalah sebagai berikut:

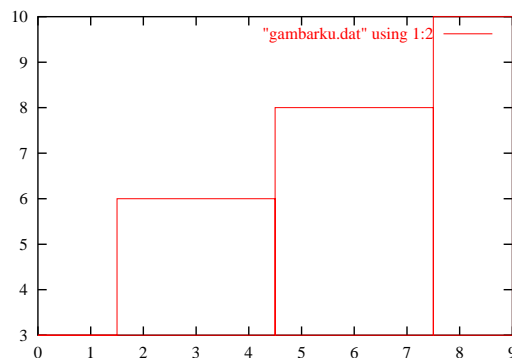
```
0 3 2 2
3 6 4 5
6 8 7 8
9 10 11 12
```

Data tersebut disimpan dalam file `gambarku.dat`. Bila kita ingin menampilkan grafik ini ke suatu terminal keluaran yang ditentukan dapat dilakukan dengan menggunakan perintah `plot` (lihat contoh berikut):

Contoh:

```
gnuplot> plot "gambarku.dat" using 1:2 with boxes
```

Adapun hasil dari perintah ini adalah seperti pada gambar di bawah ini.

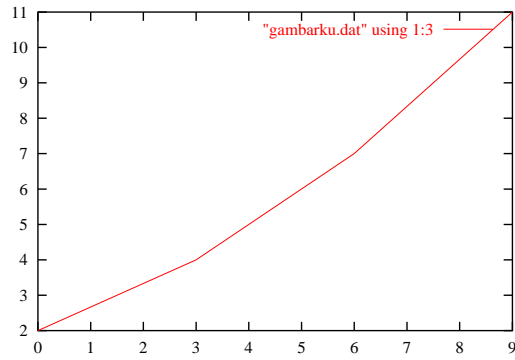


Gambar 2.1: Contoh hasil pemplotan dengan tipe boxes

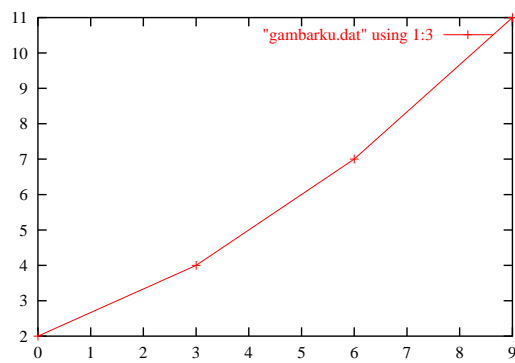
Dari file `gambarku.dat` dapat dimodifikasi beberapa macam grafik dan itu berasal dari beberapa perintah tentunya seperti contoh di bawah ini :

```
gnuplot> plot "gambarku.dat" using 1:2 with lines
gnuplot> plot "gambarku.dat" using 1:2 with linespoints
gnuplot> plot "gambarku.dat" using 1:2 with points
```

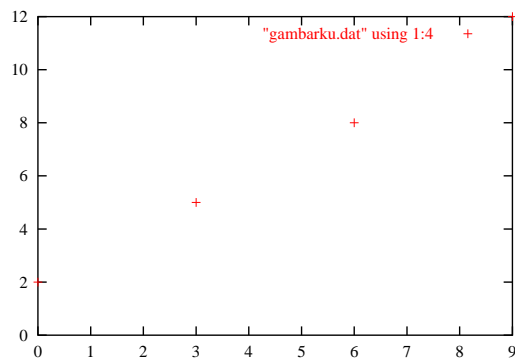
Selain memplot dengan cara pemanggilan file seperti di atas, gnuplot dapat juga memplot data yang dimasukkan secara interaktif, misalnya jika kita akan memplot data di atas untuk kolom 1:2 perintahnya adalah sebagai berikut:



Gambar 2.2: Contoh hasil pemplotan dengan tipe lines



Gambar 2.3: Contoh hasil pemplotan dengan tipe linespoints



Gambar 2.4: Contoh hasil pemplotan dengan tipe points

```
gnuplot> plot "-" with lines
0 3
3 6
6 8
9 10
<CTRL-D>
```

Dari contoh grafik di atas masih terdapat beberapa kekurangan, misalnya:

- Legenda masih default dari program gnuplot
- Keterangan sumbu X dan Y belum diset
- Ukuran grafik masih ditampilkan secara default

Penjelasan mengenai hal ini akan diuraikan pada bab-bab selanjutnya.

2.3 Menyimpan tampilan grafik dalam format lain

Apabila dari tampilan grafik yang dihasilkan akan digunakan untuk melengkapi suatu tulisan atau tujuan lainnya, tentunya tampilan grafik tersebut harus dapat disimpan dalam format grafik tertentu tergantung keperluan. Gnuplot menyediakan bermacam-macam terminal keluaran yang bisa digunakan untuk menyimpan file dalam format grafik, misalnya `postscript`, `cgm`, `hpgl`, `cdw`, `gif`, `png`, `pnm`, `TeX`, dan sebagainya. Jika suatu gambar akan digabungkan ke dalam teks atau gambar dengan kualitas cetak tinggi, format keluaran jenis `postscript` atau `TeX` sangat bagus digunakan. Perintah untuk menampilkan file dalam format lain adalah `set terminal` dan `set output`. Sebagai contoh untuk keluaran jenis `postscript`, perintahnya adalah sebagai berikut:

```
gnuplot> set term postscript landscape color "Roman" 12
gnuplot> set output "gambarku.ps"
gnuplot> plot "gambarku.dat" using 1:2 with lines
```

Jika perintah ini dijalankan maka di layar monitor tidak akan ada tampilan apa-apa. Namun untuk melihat hasil tampilan tersebut, diperlukan sebuah program yang dapat menampilkan file `postscript` (`*.ps`). Di lingkungan Linux terdapat beberapa program yang dapat menampilkan jenis file ini, contohnya adalah `PS Viewer` (KDE), `ghostview`, sedangkan di lingkungan MS Windows ada `Aladin GSVIEW`. Penjelasan dari perintah-perintah di atas adalah sebagai berikut :

- `set term postscript` digunakan untuk menentukan tampilan grafik yang akan disimpan dalam file `postscript`.
- `landscape` atau `portrait` digunakan untuk menentukan bentuk tampilan dalam `landscape` atau `portrait`.
- `color` digunakan untuk menentukan tampilan grafik dalam bentuk berwarna (default `monochrome`).
- `"Roman"` atau `"Helvetica"` digunakan untuk menentukan jenis huruf dari tulisan yang akan digunakan.
- Angka `12` menyatakan nilai besar-kecil huruf dari tulisan yang akan digunakan.
- `set output` digunakan untuk memberikan nama file yang diinginkan.

Di samping sebagai file `postscript` yang bisa digabung dengan teks pada program pengolah teks seperti `LATEX`, `Lyx` atau program pengolah gambar seperti `Xfig`, `Gimp`, `Tgif`, `gnuplot` dapat juga menghasilkan keluaran dalam format yang lain tergantung dari terminal yang dipilih. Misalnya untuk keperluan website, `gnuplot` bisa menghasilkan format `PNG` (*Portable Network Graphics*) atau `GIF` (*CompuServe Graphics Interchange Format*). Sebetulnya bisa juga dalam bentuk `JPEG`, tetapi untuk grafik jenis vektor, format ini tidak bagus karena data yang hilang ketika dilakukan kompresi saat penyimpanan. Untuk melihat macam-macam terminal yang disediakan oleh `gnuplot` dapat dilakukan dengan perintah berikut ini (keluarannya diambil beberapa contoh):

```
gnuplot> set term
Available terminal types:
  unknown  Unknown terminal type - not a plotting device
  table    Dump ASCII table of X Y [Z] values to output
  linux    Linux PC with (s)vgalib
  xlib     X11 Window System (gnulib_x11 dump)
  x11      X11 Window System
  X11      X11 Window System (identical to x11)
  aifm     Adobe Illustrator 3.0 Format
  cgm      Computer Graphics Metafile
  corel    EPS format for CorelDRAW
  fig      FIG 3.1 graphics language: X graphics editor
  postscript  PostScript graphics language [mode "fontname" font_size]
  tgif     TGIF X11 [mode] [x,y] [dashed] ["font" [fontsize]]
```

2.4 Latihan

1. Buatlah grafik fungsi $f(x)=\sin(x)\cos(x)$ dengan rentang $-3.14 < x < 3.14$.
2. Simpan file grafik pada soal no. 1 dalam format postscript.
3. Buatlah sembarang data seri 8 pasang, dan simpanlah dalam file `latihan2.dat`. Buatlah grafik tipe lines, linespoints, dan points dari file `latihan2.dat`.
4. Simpan file grafik pada soal no. 3 dalam format PNG atau GIF.

Catatan: untuk melihat bagaimana sintaks set terminal png atau gif, panggil teks bantu on-line dengan perintah `help set term png` atau `help set term gif` pada prompt gnuplot.

Bab 3

Membuat grafik multiplot dan errorbar

3.1 Multiplot dalam satu grafik

Bila perintah-perintah sederhana yang dijelaskan pada bab 2 di atas digabungkan menjadi satu perintah, maka akan dihasilkan sebuah multigrafik. Hanya untuk mendapatkan sebuah tampilan yang lebih bagus sesuai dengan keinginan kita, perlu ditambahkan beberapa perintah seperti di bawah ini.

```
gnuplot> set xrange [-2:11]
gnuplot> set yrange [0:16]
gnuplot> set xlabel "Waktu kultur (hari)"
gnuplot> set ylabel "Biomasa (gram)"
gnuplot> plot "gambarku.dat" u 1:2 t "kolom 1:2" with boxes,\
> "gambarku.dat" u 1:3 t "kolom 1:3" with lines,\
> "gambarku.dat" u 1:4 t "kolom 1:4" with points
```

Bandingkan dengan perintah berikut ini:

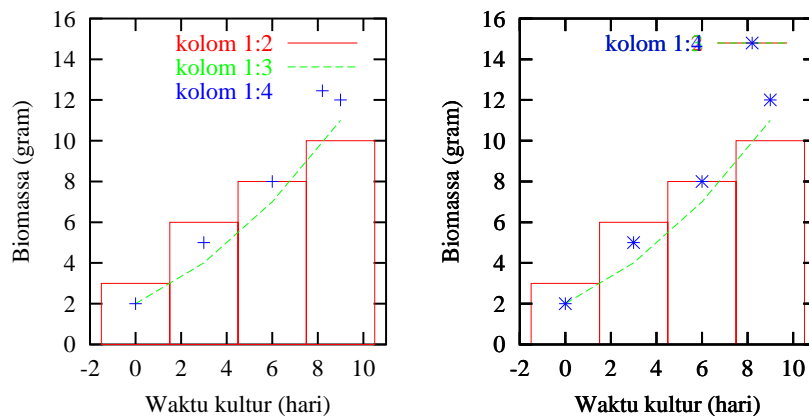
```
gnuplot> set multiplot
gnuplot> set size 1.0,1.0
gnuplot> set origin 0.0,0.0
gnuplot> set xrange [-2:11]
gnuplot> set yrange [0:16]
gnuplot> set xlabel "Waktu kultur (hari)"
gnuplot> set ylabel "Biomasa (gram)"
gnuplot> plot "gambarku.dat" u 1:2 t "kolom 1:2" with boxes
gnuplot> plot "gambarku.dat" u 1:3 t "kolom 1:3" with lines 2
gnuplot> plot "gambarku.dat" u 1:4 t "kolom 1:4" with points 3
gnuplot> set nomultiplot
```

Dari perintah di atas akan ditampilkan hasil multiplot dalam satu grafik seperti tampak pada gambar berikut ini.

Perbedaan kedua grafik ini terletak pada pemakaian perintah `set multiplot` pada grafik b (kanan). Perbedaan kedua susunan perintah di atas adalah pada grafik a (kiri) hanya grafik plot-nya saja yang diplot pada lapis yang berbeda, sedangkan pada grafik b baik grafik plot maupun sumbu dan keterangannya diplot berulang-ulang pada lapis yang berbeda. Perbedaannya dapat dilihat pada grafik b legenda menjadi saling tumpang-tindih begitu juga sumbu yang menyebabkan teks keterangan sumbu menjadi lebih tebal (seperti tercetak tebal), tetapi jika dicetak tidak tampak.

Penjelasan dari beberapa perintah dari tampilan grafik multiplot seperti di atas adalah sebagai berikut :

- `set multiplot` di baris awal dan `set nomultiplot` di baris terakhir adalah untuk menunjukkan agar pemplotan dibuat untuk banyak plot (multiplot).
- `set size 1.0,1.0` untuk menentukan ukuran 1x1 tampilan pemplotan di daerah keluaran grafik.
- `set origin 0.0,0.0` digunakan untuk menentukan koordinat (posisi) pemplotan pada daerah keluaran grafik.



Gambar 3.1: Tampilan hasil pemplotan multigrafik

- `set xrange [-2:11]` untuk mengatur kisaran sumbu X dari grafik yang kita inginkan. Untuk membuat kisaran ini perlu diperhatikan dan penyesuaian dari file `gambarku.dat`.
- `set yrange [0:16]` untuk menentukan kisaran sumbu Y dari hasil pemplotan file `gambarku.dat`. Oleh karena itu dalam membuat angka kisaran ini harus dilihat pada file `gambarku.dat`.
- `set ylabel "Biomassa (gram)"` digunakan untuk menuliskan keterangan nilai-nilai sumbu y. Penulisan keterangannya dapat dilakukan di antara tanda petik tunggal [''] atau tanda petik ganda ["].
- `set xlabel "Waktu kultur (hari)"` digunakan untuk menuliskan keterangan nilai-nilai sumbu x. Penulisan keterangannya dapat dilakukan diantara tanda petik tunggal [''] atau tanda petik ganda ["].
- `Plot "gambarku.dat" using 1:2 with boxes,\` digunakan untuk menampilkan sebuah pemplotan data yang kita miliki. Perintah plot hanya memplot satu macam grafik data artinya satu perintah plot mewakili satu halaman kertas atau sebuah grafik. Kalau ingin memplot lebih dari satu macam grafik data dalam sebuah grafik, maka tiap-tiap plot harus dipisahkan dengan tanda koma (,). Tanda [\] berfungsi sebagai tanda penghubung yang menyatakan baris berikutnya masih merupakan satu baris dari perintah tersebut. Jadi artinya sama dengan tanda [-] pada pengetikan teks di akhir sebuah baris.

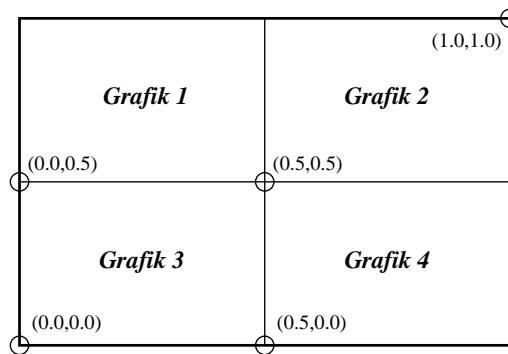
Pada tampilan pemplotan multigrafik ini dapat dilihat dengan jelas, beberapa parameter seperti keterangan sumbu X atau Y dan kisarannya telah dapat diatur sesuai dengan keinginan kita sendiri. Bila sedikit berani memodifikasi perintah tersebut, misalnya nilai kisaran sumbu X dan Y beserta keterangannya ataupun penentuan ukuran dan koordinatnya, maka akan diperoleh sebuah tampilan multigrafik yang sesuai dengan keinginan kita..

3.2 Multiplot dalam satu halaman kertas

Ada kalanya, karena suatu alasan tertentu, maka beberapa plot data yang ada, harus ditampilkan dalam satu halaman kertas atau satu keluaran gambar. Hal ini berarti bahwa dalam satu halaman kertas bisa ditampilkan dua atau lebih grafik (lihat susunan grafik pada gambar berikut). Untuk membuat grafik seperti ini berdasarkan perintah-perintah yang telah diuraikan di atas sudah dapat dilakukan. Hanya perlu ditambahkan beberapa pengaturan ukuran dan posisi gambar.

Mari kita perhatikan contoh perintah seperti ditampilkan dalam skrip di bawah ini. Dalam hal ini dibuat file baru untuk data yang akan digunakan, misalnya `fileku.dat` yang berisi nilai-nilai seperti di bawah ini. Kolom 2, 3, 4 dan 5 mewakili pertumbuhan jenis rumput laut *Gracilaria*, *Eucheuma*, *Gelidium* dan *Sargassum*.

```
0 3 2 2 2
2 5 3 4 4
4 7 6 7 7
6 9 8 9 9
8 11 12 10 10
10 13 14 13 14
```



Gambar 3.2: Susunan empat buah grafik pada perintah multiplot

Adapun perintah yang dibutuhkan untuk menampilkan multi grafiknya adalah sebagai berikut :

```

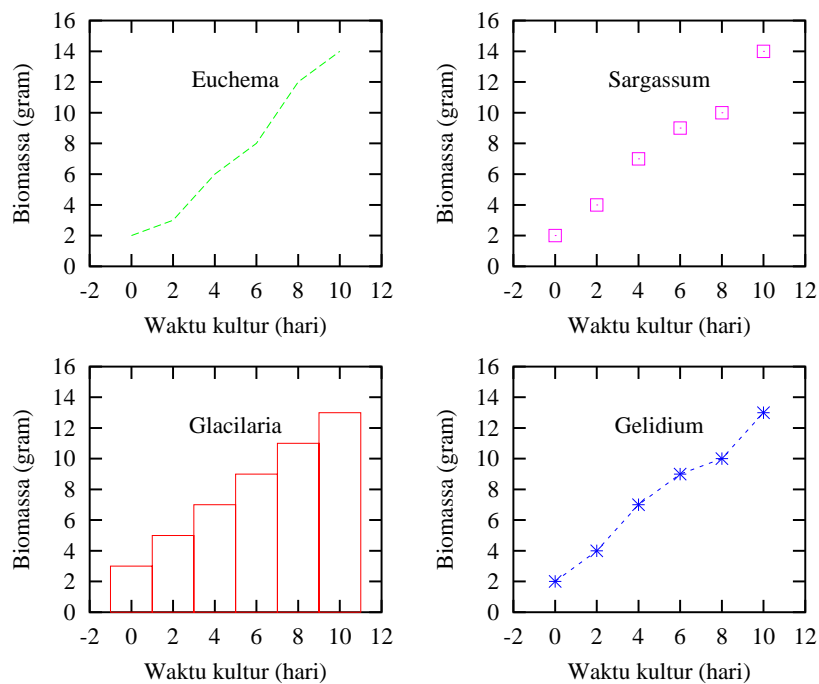
set multiplot
set size 1.0,1.0
# Grafik pertama
set size 0.5,0.5
set origin 0.0,0.0
set nokey
set xrange [-2:12]
set yrange [0:16]
set xlabel "Waktu kultur (hari)"
set ylabel "Biomasa (gram)"
set title "Gracilaria" ,-4
plot "fileku.dat" using 1:2 with boxes
# Grafik kedua
set size 0.5,0.5
set origin 0.0,0.5
set title "Eucheuma" ,-4
plot "fileku.dat" using 1:3 with lines
# Grafik ketiga
set size 0.5,0.5
set origin 0.5,0.0
set title "Gelidium" ,-4
plot "fileku.dat" using 1:4 with linespoints
# Grafik keempat
set size 0.5,0.5
set origin 0.5,0.5
set title "Sargassum" ,-4
plot "fileku.dat" using 1:5 with points
set nomultiplot
pause -1 "Tekan tombol Enter <RETURN>"

```

Sedangkan penampilan grafik dari perintah di atas adalah seperti pada gambar berikut ini.

Kalau diperhatikan dengan seksama format perintah multigrafik di atas ini terdapat beberapa parameter yang perlu dirubah, yaitu :

- Setiap pemplotan mempunyai format ukuran (`set size`) dan posisi atau koordinat (`set origin`) grafik yang ditampilkan. Bila dalam satu halaman akan ditampilkan 4 buah grafik, maka perintahnya `set size 0.5,0.5`. Perintah ini diberikan ke tiap-tiap grafik. Begitu juga untuk format posisi atau koordinat grafik yang ditampilkan, maka perintahnya `set origin 0.0,0.0; 0.0,0.5; 0.5,0.0` dan `0.5,0.5`.
- Perintah `set nokey` digunakan untuk menghapus default legenda yang dibuat oleh gnuplot
- Pada perintah `set title` berguna untuk menambahkan judul pada tiap-tiap grafik



Gambar 3.3: Tampilan hasil pemplotan multigrafik dalam satu halaman kertas

Berbekal dari uraian di atas, maka bila beberapa contoh perintah dimodifikasi maka sudah dapat dihasilkan beberapa variasi grafik dalam berbagai bentuk tampilan, misal multiplot dari multi grafik dalam satu halaman kertas dengan atau tanpa label di sumbu X atau Y, dan lain-lain. Dan bila menginginkan jenis tampilan grafik lainnya misalnya grafik linier dengan standar deviasi, dan lain-lain dapat dibaca manual `gnuplot` yang ditampilkan di internet atau secara on-line.

3.3 Membuat grafik errorbars

Kadang kala dalam sebuah penelitian atau kegiatan survai dilakukan pengambilan data dengan jumlah pengulangan tertentu untuk mendapatkan suatu hasil dengan tingkat ketelitian tertentu. Untuk menyajikan hasil penelitian tersebut diperlukan sebuah grafik dengan standar deviasi yang dikenal dengan nama `errorbars`.

Tipe grafik `errorbars` merupakan salah satu jenis tipe grafik yang ada dalam program `gnuplot`. Tipe ini digunakan untuk menggambarkan besarnya error (kesalahan) atau deviasi dari suatu titik-titik plot. Dan program `gnuplot` untuk menampilkan grafik jenis ini tidaklah sulit. Adapun tipe `errorbars` yang disediakan oleh program `gnuplot` terdapat beberapa jenis, yaitu `xerrorbars`, `yerrorbars` (atau `errorbars`), `xyerrorbars`, `boxerrorbars` dan `boxes`.

Untuk menampilkan grafik jenis ini, pada program `gnuplot` dibutuhkan sebuah file data berbentuk ASCII yang mempunyai susunan minimum terdiri dari 3 kolom data, yaitu X, Y dan delta-Y (menyatakan sebagai besarnya error). Untuk lebih jelasnya diperhatikan contoh berikut ini. Dari hasil penimbangan berat basah (dengan pengulangan n) pada pertumbuhan rumput laut *Sargassum* dari Karimunjawa dan Takabonerate terhadap perubahan salinitas (kadar garam) diperoleh data yang disimpan dalam file `error.dat` dan terdiri sebagai berikut :

```
# contoh file error.dat
# kolom : Salinitas, berat, error, berat, error, error
10 19 1.7 11 1.2 1.1
15 24 1.3 19 1.1 1.3
20 29 3.3 21 0.9 1.2
25 26 1.5 19 1.0 1.4
30 35 2.5 27 1.4 0.9
35 29 1.9 20 1.1 1.1
```

```
40 25 1.5 16 1.5 1.2
45 17 1.8 13 1.1 1.3
```

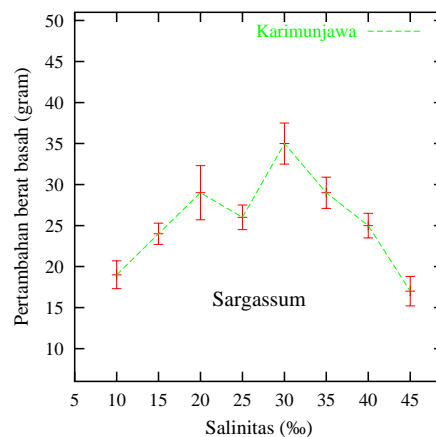
Selain data ASCII seperti tersebut di atas, diperlukan juga perintah-perintah yang mempunyai sintak sebagai berikut :

```
set function style <style>
set data style <style>
show function style
show data style
```

Berdasarkan hal tersebut, bila digabungkan dengan beberapa perintah-perintah yang telah diuraikan di atas, maka untuk menampilkan grafik errorbars dari file `error.dat`, maka diperlukan skrip sebagai berikut :

```
# Grafik Pertambahan berat Sargassum dari Karimunjawa
set size square 0.6,0.6
set data style errorbars
set xrange [5:49]
set yrange [6:51]
set xlabel "Salinitas (o/oo)"
set ylabel "Pertambahan berat basah (gram)"
set bars large
set title "Sargassum" ,-10
plot "errorku.dat" u 1:2:3 notitle with errorbars,\
    "" u 1:2 title "Karimunjawa" with lines l
pause -1 "Tekan tombol Enter <RETURN>"
```

Adapun tampilan gambar dari skrip di atas dapat dilihat seperti gambar dibawah ini. Grafik ini merupakan hasil dari sebuah plot contoh *Sargassum* dari Karimunjawa dan dari file `error.dat` diperoleh dari kolom 1, 2 dan 3.



Gambar 3.4: Tampilan grafik tipe errorbars

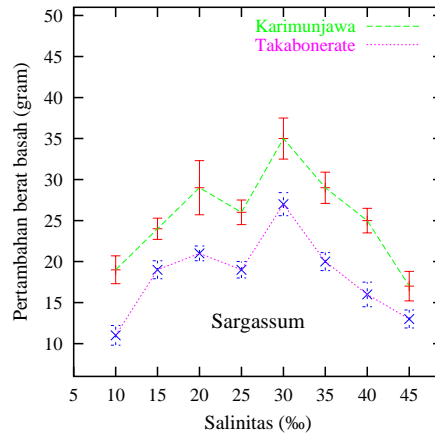
Kadang kala kegiatan pengambilan data tidak hanya dilakukan dalam satu tempat saja, sehingga data yang diperoleh menjadi semakin banyak. Dan bila diamati dari file `error.dat` juga dapat dikembangkan menjadi dua beberapa grafik. Untuk mewujudkan hal ini dapat diperhatikan skrip di bawah ini, yaitu :

```
set size square 0.6,0.6
set data style errorbars
set xrange [5:49]
set yrange [6:49]
set xlabel "Salinitas (o/oo)"
set ylabel "Pertambahan berat basah (gram)"
set bars large
set title "Sargassum" ,-12.5
plot "errorku.dat" u 1:2:3 notitle with errorbars, \
```

```

"" u 1:2 t "Karimunjawa" with lines, \
"" u 1:4:5 notitle with errorbar, \
"" u 1:4 t "Takabonerate" with lines
pause -1 "Tekan tombol Enter <RETURN>"
    
```

Berdasarkan skrip di atas dapat diperoleh sebuah grafik dengan dua buah plot dari 2 lokasi pengambilan data, yaitu :



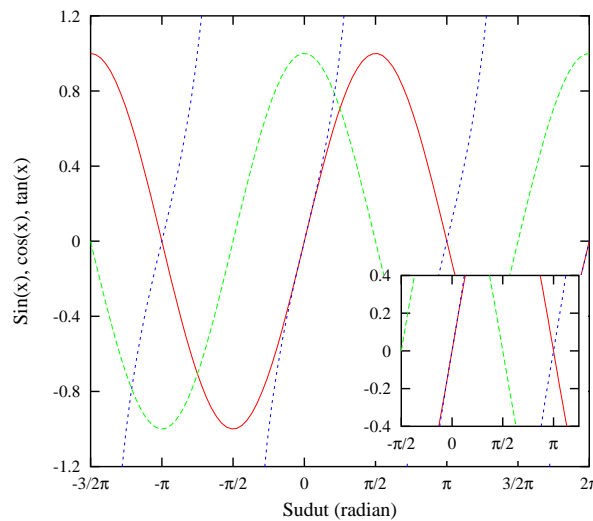
Gambar 3.5: Tampilan grafik hasil multiplot tipe errorbars

Dari dua buah contoh grafik errorbars tersebut, bila dilihat secara cermat pada skripnya tidak banyak perbedaannya. Pada skrip kedua terdapat penambahan 2 baris perintah. Dan dari file `error.dat` masih bisa dikembangkan menjadi beberapa grafik tipe yang lain, misalnya `xerrorbar`, `xyerrorbar`, `boxes`, dan lain-lain.

Untuk menambah wawasan akan kegunaan gnuplot dan kehandalannya dalam menampilkan grafik dapat diikuti uraian dalam bab selanjutnya. Sesuatu hal yang perlu diperhatikan pada saat menjalankan gnuplot adalah memperhatikan dengan seksama penulisan perintahnya, kesalahan yang timbul biasanya disebabkan hanya kesalahan penulisan.

3.4 Latihan

1. Buatlah grafik multiplot yang terdiri dari 6 grafik, masing-masing: $\sin(x)$, $\cos(x)$, $\tan(x)$, $f(x)=2x+1$, $f(x)=x^2+2x+1$, dan $f(x)=\sin(x)*\cos(x)$.
2. Buatlah skrip gnuplot yang menghasilkan grafik seperti dibawah ini:



3. Ubahlah tipe grafik dari contoh errorbar di atas, misalnya menjadi `xerrorbar`, `xyerrorbars`, `boxerrorbars`, `boxxyerrorbars`.
4. Dari empat grafik ini gabungkanlah sehingga diperoleh sebuah grafik multiplot.

Bab 4

Ekstrapolasi

Pada bagian ini akan diterangkan kemampuan kurva fitting dari program `gnuplot`. Sepengetahuan penulis, `gnuplot` dapat melakukan 2 macam tipe fitting dengan menggunakan suatu implementasi yang dikenal dengan nama algoritma non-linear kuadrat terkecil (NLLS=*Non-Linear Least Square*) Marquardt-Lavenberg. Tipe pertama adalah regresi atau fitting linear dimana optimasi dilakukan pada suatu persamaan linear atau suatu persamaan yang dapat dinyatakan dalam suatu bentuk linear. Ini termasuk fitting polinomial dan bentuk-bentuk persamaan tertentu. Tipe yang lain adalah non-linear dan memberikan kebebasan kepada pengguna untuk menentukan sendiri persamaannya. Pada fitting non-linear, `gnuplot` memberikan kebebasan kepada pengguna untuk menentukan persamaan yang akan digunakan secara bebas.

4.1 Melakukan fitting data

Dalam program `gnuplot` ada dua hal yang harus dilakukan untuk melakukan fitting suatu data, yaitu membuat suatu definisi fungsi kecenderungan data dalam sebuah file, misalnya $f(x) = y_0 + m \cdot x$ (fungsi linear). Fungsi ini disimpan dalam file misalnya `line.fnc`. Membuat skrip yang akan dikerjakan oleh `gnuplot` (lihat sintaks dan contoh berikut).

Sintaks:

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

Contoh pemakaian:

```
f(x) = a*x**2 + b*x + c
g(x,y) = a*x**2 + b*y**2 + c*x*y
FIT_LIMIT = 1e-6 fit f(x) 'measured.dat' via 'start.par'
fit f(x) 'measured.dat' using 3:($7-5) via 'start.par'
fit f(x) './data/trash.dat' using 1:2:3 via a, b, c
fit g(x,y) 'surface.dat' using 1:2:3:(1) via a, b, c
```

Misalkan data yang akan dicari fit-nya disimpan pada file `ekstra.dat` sebagai berikut:

```
# file ekstra.dat
# [1]porositas [2]ukuran butir
# data yang dipakai pada contoh grafik tidak ditulis
# semuanya, tapi cukup untuk digunakan mencoba skrip
#
14.482  0.279
 9.245  0.247
14.992  0.282
12.320  0.264
14.322  0.278
```



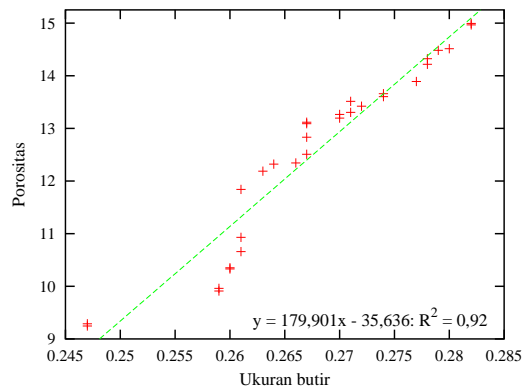
```

13.264  0.270
11.840  0.261
13.194  0.270
12.187  0.263
10.330  0.260
 9.962  0.259
14.966  0.282
10.658  0.261
13.117  0.267
    
```

Contoh skrip gnuplot untuk menghitung nilai fitting data di atas adalah sebagai berikut:

```

# contoh fitting
set xlabel "Ukuran butir"
set ylabel "Porositas"
set xrange [0.245:0.285]
set yrange [9:15.25]
set xtics 0.245,0.005
set ytics 9,1 set nokey
set nokey
load 'line.fnc', y0 = 0.0, m = 0.0
fit l(x) 'ekstra.dat' u 2:1 via y0, m
plot 'ekstra.dat' u 2:1, l(x)
pause -1 "Tekan tombol Enter <RETURN>"
reset
    
```



Gambar 4.1: Contoh ekstrapolasi linear dengan fungsi $f(x) = y_0 + m \cdot x$

Perintah `load 'line.fnc'` akan memanggil fungsi kecenderungan yang telah dibuat pada file `line.fnc` dan akan diberi nilai awal $y_0 = 0.0$ dan $m = 0.0$. Perintah `fit l(x)` akan melakukan fitting terhadap data berdasarkan fungsi yang terdapat pada file tersebut hingga nilai batas fit dipenuhi (default `fit_limit = 1e-5`).

4.2 Latihan

1. Buatlah skrip gnuplot untuk memplot contoh data pada bab 2.1 dengan tipe `linespoints`
2. Buatlah skrip gnuplot untuk mencari fungsi kecenderungan dari data soal no. 1 di atas.
3. Ubahlah fungsi pada file `line.fnc` ke dalam fungsi kuadrat (power).

Bab 5

Data multivariat

5.1 Grafik 3 dimensi

Untuk memplot data 3 dimensi, `gnuplot` hanya mendukung koordinat Cartesius. `splot` merupakan perintah untuk menggambar pemplotan 3 dimensi yang dapat membuat pemplotan suatu fungsi atau file data yang mirip sekali dengan perintah `plot`.

Sintaks `splot`:

```
splot {<ranges>}
      <function> | "<datafile>" {datafile-modifiers}}
      {<title-spec>} {with <style>}
      {, {definitions,} <function> ...}
```

Sintaks untuk memplot file data:

```
splot '<file_name>' {binary | matrix}
      {index <index list>}
      {every <every list>}
      {using <using list>}
```

Contoh pemakaian

```
set xlabel "tars1"
set ylabel "tars2"
set zlabel "head"
splot "flea.dat" using 1:2:3 title "Group 1", \
      "" using 4:5:6 title "Group 2", \
      "" using 7:8:9 title "Group 3"
pause -1 "Tekan tombol Enter <RETURN>"
reset
```

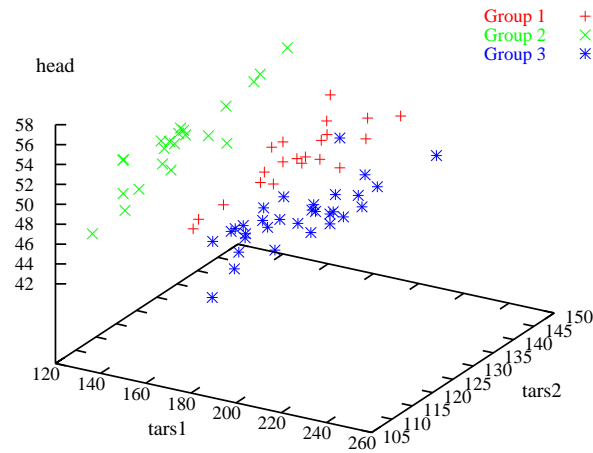
Isi file `flea.dat` (data diambil dari contoh pada program `Xgobi`):

```
191 131 53 158 141 58 186 107 49
185 134 50 146 119 51 211 122 49
200 137 52 151 130 51 201 114 47
173 127 50 122 113 45 242 131 54
171 118 49 138 121 53 184 108 43
160 118 47 132 115 49 211 118 51
188 134 54 131 127 51 217 122 49
186 129 51 135 123 50 223 127 51
174 131 52 125 119 51 208 125 50
163 115 47 130 120 48 199 124 46
190 143 52 130 131 51 211 129 49
```

```

174 131 50 138 127 52 218 126 49
201 130 51 130 116 52 203 122 49
190 133 53 143 123 54 192 116 49
    
```

Hasil pemplotan file di atas adalah sebagai berikut:



Gambar 5.1: Contoh hasil pemplotan data 3 dimensi

Mengubah sudut pandangan dengan `set view`

Sintaks:

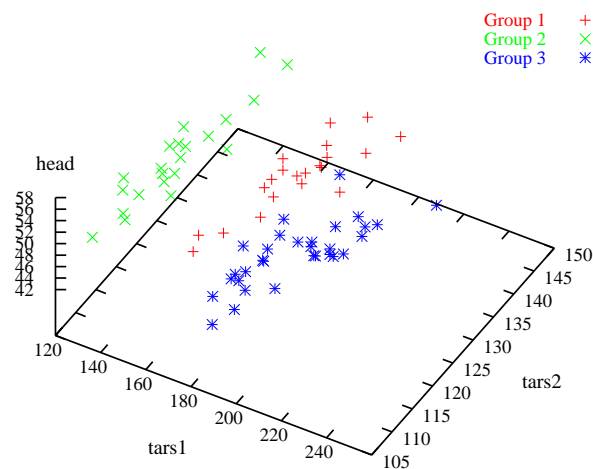
```

set view <rot_x> {, {<rot_z>} {, {<scale>} {, <scale_z>}}
show view
    
```

Contoh:

```

# merubah sudut pandangan
set view 30,30,1,1
replot
pause -1 "Tekan tombol Enter <RETURN>"
    
```



Gambar 5.2: Contoh hasil perubahan sudut pandangan

5.2 Menggambar kontur

Dalam kamus geologi didefinisikan bahwa penggambaran kontur (*contouring*) merupakan pembuatan garis (kontur) yang menghubungkan titik-titik yang memiliki kesamaan elevasi pada suatu peta yang mewakili topografi (*Glossary Geology, fourth edition*). Jadi dapat juga dikatakan, pemplotan garis yang memiliki kesamaan nilai dari data 3-dimensi ke dalam grafik 2-dimensi. Dimensi yang ketiga dinyatakan oleh nilai pada garis kontur.

Dikenal ada 2 macam metode untuk membuat gambar kontur, yaitu "gridding" dan "TiN" (*triangulated irregular network*) namun yang paling populer adalah metode *gridding* (penggridan). Prinsip dari metode ini adalah membuat titik-titik baru dengan jarak yang teratur dengan menginterpolasi titik-titik yang tersebar acak. Ada enam metode intrapolasi yang biasa dipakai untuk melakukan penggridan, misalnya *triangulation*, *invers distance weighting*, *nearest neighbor*, *ordinary Kriging*, *minimum curvature*, dan *trend surface*. Pada program gnuplot hanya tersedia satu metode untuk melakukan penggridan yaitu *invers distance weighting*. Gnuplot menyediakan 2 macam cara untuk membuat kontur, yaitu memplot data yang telah digrid dan memplot data acak (belum digrid).

5.2.1 Data teratur

Pada subbab ini hanya akan dibahas mengenai pemplotan dari data yang telah digrid. Ini berlaku jika penggridan tidak diserahkan secara otomatis pada gnuplot, karena seperti ditulis di atas bahwa program ini hanya menyediakan satu macam metode untuk melakukan penggridan, sehingga untuk membuat kontur dengan metode penggridan yang berbeda dapat juga dilakukan, misalnya dengan membuat suatu skrip dengan bahasa pemrograman tertentu atau penggridan dilakukan oleh program lain, seperti *Uncert* yang memiliki fasilitas penggridan paling lengkap di lingkungan Linux.

Pada program gnuplot pembuatan gambar kontur dilakukan oleh 3 perintah utama yaitu `set contour`, `set cntrparam` dan `plot`.

Sintaks dari perintah ini adalah sebagai berikut:

```
set contour {base | surface | both}
set nocontour
show contour
```

Sedangkan perintah `set cntrparam`:

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

Perintah ini memiliki dua fungsi. Pertama, menset nilai z dimana titik kontur ditentukan misalnya oleh interpolasi antara titik data atau fungsi `isosample`. Kedua, mengontrol cara penggambaran kontur antara titik-titik yang ditentukan berdasarkan kesamaan z . `<n>` harus berupa pernyataan konstanta integral dan `<z1>`, `<z2>` ... menyatakan suatu konstanta.

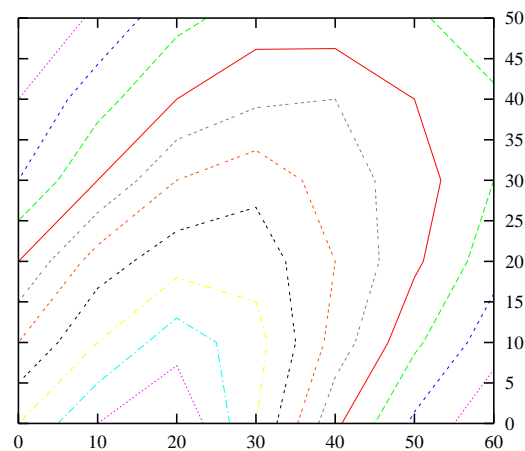
'linear', 'cubicspline', 'bspline' – mengontrol tipe pendekatan atau interpolasi pada penggambaran kontur. Jika 'linear', segmen garis lurus menghubungkan titik-titik yang mempunyai nilai z yang sama. Jika 'cubicspline', kontur linear akan diinterpolasi antar titik dengan nilai z yang sama untuk membentuk garis kontur yang (melengkung) halus. Jika 'bspline', dapat dipastikan bahwa garis lengkung halus akan digambar untuk menghubungkan titik-titik yang mempunyai nilai z yang sama.

Data yang akan diplot sebagai gambar kontur harus disusun seperti matrik berikut ini:

$x_{0,0}$	$y_{0,0}$	$z_{0,0}$
$x_{0,1}$	$y_{0,1}$	$z_{0,1}$
$x_{0,2}$	$y_{0,2}$	$z_{0,2}$
.	.	.
$x_{0,N}$	$y_{0,N}$	$z_{0,N}$
$x_{1,0}$	$y_{1,0}$	$z_{1,0}$
$x_{1,1}$	$y_{1,1}$	$z_{1,1}$
$x_{1,2}$	$y_{1,2}$	$z_{1,2}$
.	.	.
$x_{1,N}$	$y_{1,N}$	$z_{1,N}$

Script gnuplot untuk membuat kontur¹ dari data ini adalah sebagai berikut:

```
set xrange [0:60]
set yrange [0:50]
set nosurface
set contour
set cntrparam levels incremental 50,50,600
set view 0,0,1,1
set nokey
splot 'contour.dat' with lines 2
pause -1 "Tekan tombol Enter <RETURN>"
set cntrparam bspline # memperhalus gambar
pause -1 "Tekan tombol Enter <RETURN>"
```



Gambar 5.3: Gambar kontur yang diplot dari data teratur tanpa setting cntrparam

5.2.2 Data tidak teratur (acak)

Seperti diterangkan pada subbab sebelumnya, bahwa data yang bisa diplot oleh gnuplot hanya data grid, sehingga data acak harus digrid terlebih dahulu sebelum dilakukan pemplotan dengan gnuplot. Dalam program ini proses penggridan dapat dilakukan oleh perintah `set dgrid3d` (lihat sintaks berikut). Perintah ini akan mengkonversi data acak menjadi data grid yang nantinya akan diplot oleh perintah `splot`. `Dgrid3d` menggunakan metode *inverse distance weighting* untuk mencari besarnya nilai tiap-tiap titik grid (node).

Sintaks:

¹Mengingat proses pengeplotan data grid pada gnuplot tidak secara simultan, artinya gnuplot hanya membaca dan mengolah data per sekuen (tiap baris), maka pada data yang kompleks gnuplot tidak dapat menghasilkan gambar kontur yang representatif (mewakili sebaran data asli). Tetapi di dalam lingkungan Linux ada beberapa program yang dapat digunakan untuk menggambar kontur seperti di atas, misalnya GRASS, Uncert, GMT, dan Gri. Sedangkan untuk mengeplot persamaan 3-D dengan program gnuplot sudah cukup, atau bisa juga dengan program Scigraphica atau R-statistics.

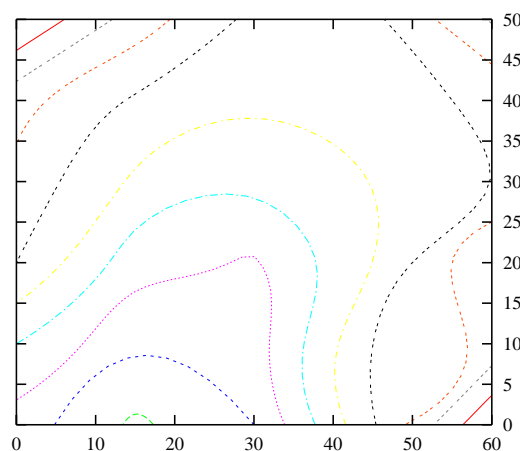
```

set dgrid3d {<row_size>} {, {<col_size>} {, {<norm>}}
set nodgrid3d show nodgrid3d
show nodgrid3d

```

Titik-titik grid yang dibuat ditentukan oleh parameter `row_size`, `col_size` dan `norm` yang menyatakan faktor pemberat (*weighting factor*). Pada metode IDS (*invers distance square*) yang cukup populer parameter `norm` diberi nilai 2 (lihat contoh berikut). Data yang akan diplot sebagai gambar kontur harus disusun seperti matrik berikut ini:

$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,1}$	$a_{3,2}$	$a_{3,3}$
$a_{4,1}$	$a_{4,2}$	$a_{4,3}$
.	.	.
.	.	.
.	.	.
$a_{N,1}$	$a_{N,2}$	$a_{N,3}$



Gambar 5.4: Gambar kontur yang dihasilkan dari data acak dengan seting `cntrparam cubicspline`.

Script gnuplot untuk membuat gambar kontur di atas adalah sebagai berikut:

```

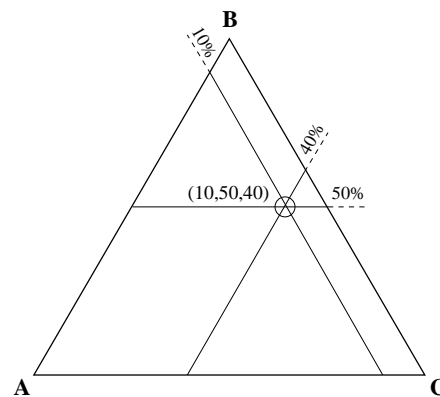
set xrange [0:60]
set yrange [0:50]
set dgrid3d 6,5,2
set cntrparam levels incremental 50,50,600
set nosurface
set contour
set view 0,0,1,1
set nokey
splot 'contour1.dat' with lines
pause -1 "Tekan tombol Enter <RETURN>"
# dengan smoothing cubicspline
set cntrparam cubicspline
replot
pause -1 "Tekan tombol Enter <RETURN>"

```

5.3 Menggambar diagram segitiga

Diagram segitiga atau *ternary diagram* merupakan diagram untuk memplot proporsi komposisi dari suatu material yang terdiri dari 3 komponen (dinyatakan dalam persen). Diagram ini banyak digunakan pada bidang kimia, farmasi, teknik mesin (material) dan geologi baik untuk menggambarkan diagram fase suatu material atau untuk klasifikasi suatu material yang berdasarkan 3 komponen utama.

Untuk melihat bagaimana prinsip pemplotan dan pembacaan diagram segitiga, mari kita lihat gambar berikut ini.



Gambar 5.5: Prinsip pemplotan dan pembacaan diagram segitiga

Pada gambar di atas terlihat ada 3 komponen yang akan diplot yaitu A, B, dan C yang masing-masing dinyatakan dalam persen). Jumlah total 3 komponen ini harus 100%. Kalau kita baca titik yang diplot adalah $P(A,B,C) = P(10,50,40)$. Hati-hati memberikan persamaan pada gnuplot, karena posisi plot A dan C bisa terbalik!

Untuk lebih jelasnya silakan amati contoh berikut ini:

Contoh perintah gnuplot

```
set size square 0.535,0.65
set angle degrees
set noborder
set xrange [-10:100]
set yrange [-10:100]
set noxtics
set noytics
set key 100,90

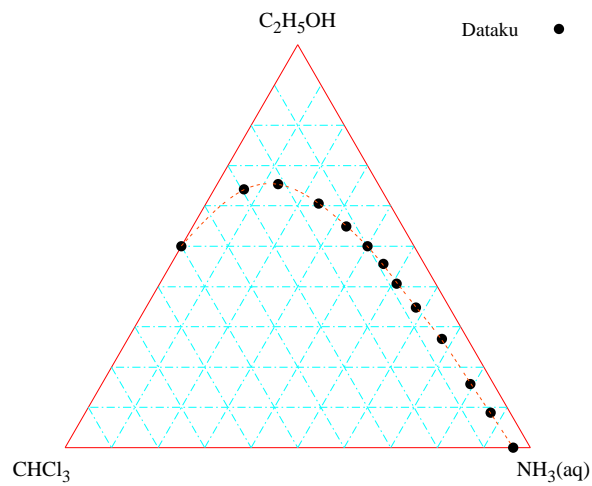
gx(x,y) = x + y * cos(60)
gy(x,y) = y * cos(30)

# Memplot batas segitiga, jaring-jaring dan data
# Dari sini perubahan hasil dapat dilakukan sesuai keinginan

set label 41 "CHCl_3" at -5,-5 center # label x-Axis
set label 42 "C_2H_5OH" at 50,92.5 center # label y-Axis
set label 43 "NH_3(aq)" at 105,-5 center # label z-Axis
plot "tern-bnd.dat" u (gx($1,$2)):(gy($1,$2)) notitle with lines 1, \
    "ter-mesh.dat" using (gx($1,$2)):(gy($1,$2)) notitle with lines 5, \
    "ternary.dat" u (gx($1,$2)):(gy($1,$2)) t "Dataku" with points 7, \
    "ternary.dat" u (gx($1,$2)):(gy($1,$2)):(0.1) smooth acsplines notit-
tle with lines 8
pause -1 "Tekan tombol Enter <RETURN>"
```

Penjelasan dari skrip di atas adalah sebagai berikut:

- Fungsi $gx(x,y)$ dan $gy(x,y)$ berguna untuk mengkonversi data terner yang telah dinormalisasi (total 100%) ke dalam koordinat Cartesius dari diagram segitiga.
- `set size square 0.535,0.65` berguna untuk menentukan ukuran grafik dengan skala sumbu x dan y yang sama (square).
- File `tern-bnd.dat` berisi nilai koordinat batas luar diagram segitiga, sedangkan `ter-mesh.dat` berisi nilai koordinat jaring-jaring dalam segitiga. File `ternary.dat` merupakan data yang akan diplot ke dalam diagram segitiga tersebut.



Gambar 5.6: Hasil pemplotan skrip diagram segitiga

5.4 Latihan

1. Buatlah grafik scatter dari data `contour1.dat` (lihat lampiran B).
2. Buatlah grafik kontur dari soal no. 1 dengan menambahkan satu baris perintah `set surface`.
3. Ubah tampilan grafik soal no. 2 dengan menambahkan perintah `set nocontour`.

Bab 6

Mengatur tampilan grafik

6.1 Seting terminal PostScript

Pada subbab ini akan dibahas bagaimana menset tampilan grafik sehingga menghasilkan grafik yang bagus. Penaturan tampilan grafik pada program gnuplot sangat tergantung pada terminal keluaran yang akan dipakai. Ada beberapa terminal yang mempunyai fasilitas lengkap untuk menset tampilan grafik, misalnya postscript, latex dan gabungan keduanya. Ketiga terminal ini dipakai terutama untuk membuat grafik kualitas tinggi untuk dicetak pada printer postscript. Jika suatu grafik memuat keterangan persamaan matematik atau rumus kimia yang biasanya selalu berisi subskrip atau superskrip, maka terminal postscript jenis enhanced dan terminal latex sangat bagus digunakan.

Sintaks terminal postscript:

```
set terminal postscript {<mode>} {enhanced | noenhanced}
                        {color | monochrome} {solid | dashed}
                        {<duplexing>}
                        {"<fontname>"}{<fontsize>}
set output "<filename>"
```

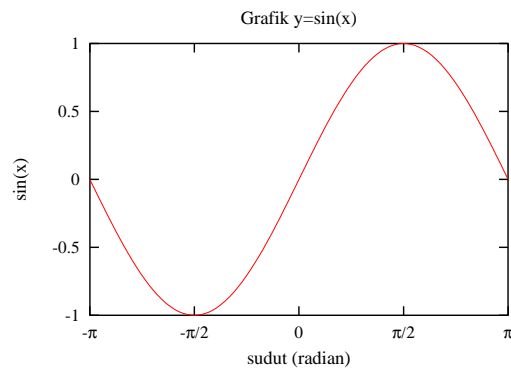
dimana <mode> meliputi 'landscape', 'portrait', 'eps' atau 'default'; 'solid' menggambar semua plot dengan garis padat, mengganti semua jenis pola garis putus-putus; <duplexing> tergantung jenis printer yang digunakan, 'duplex' akan dicetak bolak-balik sedangkan 'simplex' akan dicetak biasa; "<fontname>" adalah nama jenis huruf yang sesuai dengan huruf PostScript.

Contoh:

```
set terminal postscript default      # old postscript
set terminal postscript enhanced    # old enhpost
set terminal postscript landscape 22 # old psbig
set terminal postscript eps 14      # old epsfl
set terminal postscript eps 22      # old epsf2
set size 0.7,1.4; set term post portrait color "Times-Roman" 14
```

Penerapannya lihat contoh di bawah ini.

```
set term postscript landscape "Roman" 14
set output "tampilan1.ps"
set title "Grafik y=sin(x)"
set xlabel "sudut (radian)"
set ylabel "sin(x)"
set nokey
plot [-3.14:3.14] sin(x)
```



Gambar 6.1: Pengesetan terminal postscript

6.2 Menambahkan teks dan panah pada grafik

Seringkali legenda pada suatu gambar atau grafik tidak memberikan informasi yang cukup kepada pembaca. Biasanya tambahan keterangan seperlunya diperlukan untuk memperjelas suatu grafik, misalnya dengan menambahkan label dan penunjuk panah. Untuk menambah dua keterangan ini tidak terlalu sulit dilakukan dengan program `gnuplot`. Ada 2 perintah `gnuplot` yang dapat melakukan kedua hal ini, yaitu `set label` dan `set arrow`. Mari kita lihat sintaks dan contoh berikut ini:

Sintaks `set label`

```
set label {<tag>} {"<label_text>"} {at <position>}
           {<justification>} {{no}rotate} {font "<name><,size>"}
set nlabel {<tag>}
show label
```

Tag merupakan nilai integer yang digunakan untuk menentukan nomor atau tanda label. Jika tag tidak didefinisikan, nilai tag terendah yang tidak digunakan akan ditentukan secara otomatis oleh `gnuplot`. Tag dapat digunakan untuk menghapus atau memodifikasi label tertentu pada grafik. Untuk merubah atribut tertentu dari suatu label, gunakan perintah `set label` sesuai dengan nomor tag.

Contoh:

Untuk mengeset suatu label pada (1,2) untuk "y=x", gunakan:

```
set label "y=x" at 1,2
```

Untuk menset suatu Sigma dengan ukuran 24, dari set font Symbol, pada tengah-tengah suatu grafik, gunakan:

```
set label "S" at graph 0.5,0.5 center font "Symbol,24"
```

Untuk menset label "y=x^2" dengan posisi rata kanan pada (2,3,4), dan tag dari label sebagai nomor 3, gunakan:

```
set label 3 "y=x^2" at 2,3,4 right
```

Sintaks `set arrow`

```
set arrow {<tag>} {from <position>} {to <position>} {{no}head}
           { {linestyle | ls <line_style>}
             | { {linetype | lt <line_type>}
               { {linewidth | lw <line_width>}
             }
set noarrow {<tag>}
show arrow
```

Contoh:

Untuk menset tanda panah dari titik awal ke titik (1,2) dengan tipe garis nomor 5, gunakan:

```
set arrow to 1,2 ls 5
```

Untuk menset anak panah dari kiri bawah daerah plot ke (-5,5,3), dan nomor tag panah 3, gunakan:

```
set arrow 3 from graph 0,0 to -5,5,3
```

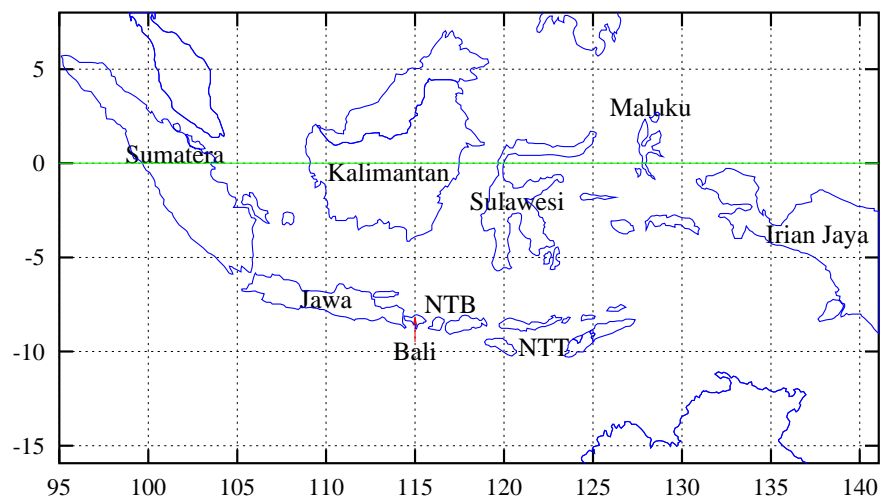
Contoh pemakaian

```
# Contoh plot peta Indonesia dan negara tetangga dengan gnuplot
set size ratio .55 0.6,0.475
set xrange [95:141.065]
set yrange [-15.9343:8.00469]
set nokey set nozeroaxis
set grid

# menambah label
set label "Sumatera" at 101.5, 0.5 center
set label "Kalimantan" at 113.5, -0.5 center
set label "Sulawesi" at 120.75, -2 center
set label "Maluku" at 128.25, 3 center
set label "Irian Jaya" at 140.5, -3.8 right
set label "Jawa" at 110, -7.25 center
set label "Bali" at 115, -10 center
set label "NTB" at 117, -7.5 center
set label "NTT" at 122.25, -9.75 center

# menambah panah
set arrow from 115, -9.5 to 115, -8.25

# plot indonesian map
plot 'indo.dat' with lines 3 4, 0
pause -1 "Tekan tombol Enter <RETURN>"
```



Gambar 6.2: Gambar peta Indonesia dan negara tetangga

6.3 Menset ukuran grafik

Pengaturan ukuran grafik adalah salah satu hal yang penting pada tampilan grafik, karena merupakan salah satu unsur dari suatu komposisi grafik. Perubahan ukuran keluaran grafik dari program gnuplot dilakukan dengan perintah `set size` (lihat sintaks berikut ini):

Sintaks:

```
set size {{no}square | ratio <r> | noratio} {<xscale>,<yscale>}
show size
```

Nilai `<xscale>` dan `<yscale>` adalah faktor skala untuk ukuran grafik termasuk grafik dan batas tepi. `ratio` menyebabkan gnuplot untuk mencoba membuat grafik dengan aspek rasio `<r>` (rasio panjang sumbu y dengan sumbu x) dengan proporsi yang telah ditentukan pada `<xscale>` dan `<yscale>` (lihat contoh di atas).

Arti dari nilai negatif `<r>` berbeda, jika `<r>=-1`, gnuplot akan menset skala sehingga sumbu x dan sumbu y memiliki unit panjang yang sama (sangat cocok untuk data geografi). Jika `<r>=-2`, unit pada sumbu y mempunyai 2 kali panjang dari unit sumbu x . 'square' sama dengan 'ratio 1'. Baik 'noratio' maupun 'nosquare' akan mengembalikan grafik ke aspek rasio default dari terminal yang digunakan tetapi tidak mengembalikan `<xscale>` atau `<yscale>` ke nilai defaultnya (1.0). 'ratio' dan 'square' tidak memiliki efek pada pemplotan 3-dimensi.

Contoh:

Untuk menentukan ukuran ke ukuran normal gunakan:

```
set size 1,1
```

Untuk membuat grafik dengan ukuran setengah dari ukuran normal dan bujursangkar gunakan:

```
set size square 0.5,0.5
```

Untuk membuat grafik dengan ukuran tinggi 2 kali lebar gunakan:

```
set size ratio 2
```

6.4 Mengatur lebar margin

Pengaturan lebar margin diperlukan jika gambar grafik tidak terletak pada posisi yang proporsional terhadap kedudukan bingkai gambar. Hal ini sangat penting ketika gambar akan digabungkan dengan suatu teks atau tulisan atau laporan. Tanpa pengesetan margin, posisi gambar kadang terpotong oleh bingkai gambar (terutama pada bagian label sumbu), sehingga menjadikan margin pada bagian atas gambar menjadi terlalu lebar.

Pada program gnuplot telah tersedia suatu perintah untuk mengatur lebar margin yaitu dengan perintah `set margin` (lihat sintaks berikut).

Sintaks:

```
set bmargin {<margin>}
set lmargin {<margin>}
set rmargin {<margin>}
set tmargin {<margin>}
show margin
```

Unit `<margin>` menyatakan nilai dari tinggi atau lebar huruf. Nilai positif unit ini menyatakan nilai mutlak dari margin, sedangkan nilai negatif (atau tanpa nilai) menyebabkan gnuplot menggunakan nilai terhitung. Biasanya margin dari suatu grafik secara otomatis dihitung berdasarkan *tics*, label *tic*, label sumbu, judul grafik, *timestamp* dan ukuran dari legenda jika diletakkan di luar batas tepi.

6.5 Latihan

1. Berikanlah nilai batas tepi pada skrip bab 6.1 di atas.
2. Ubahlah posisi letak judul grafik menjadi di tengah-tengah bingkai grafik dengan menambahkan perintah `set title {"<title-text>"} {<xoff>} {<yoff>} {",{<size>}"}`.
3. Cobalah ubah jenis huruf, misalnya dari Roman ke Helvetica atau Courier.

Bab 7

Menggunakan gnuplot untuk L^AT_EX

Pemakaian gnuplot untuk menghasilkan grafik dalam format L^AT_EX tidak menggunakan gnuplot secara interaktif. Dalam artian, untuk menjalankan program ini tidak dibutuhkan suatu terminal grafik karena keluarannya langsung berupa file yang kemudian dapat dikirim ke dalam suatu devais keluaran tertentu. Sebagai contoh, keluaran dari tipe terminal unix plot barangkali dapat dikirim ke suatu program yang dapat menginterpretasi standard pemplotan Unix. Tipe terminal imagen dan postscript barangkali digunakan untuk keluaran pada printer yang mengenal bahasa ini. Tipe terminal fig yang keluarannya berupa kode FIG yang dapat dibaca oleh program grafik Fig dan tipe terminal latex akan menghasilkan output dengan format T_EX yang diterjemahkan ke dalam bentuk yang dapat digunakan baik untuk dokumen T_EX maupun L^AT_EX.

Sintaks:

```
set terminal latex | emtex {courier | roman | default} {<fontsize>}
```

Contoh:

Tentang penempatan label teks:

Menggunakan default gnuplot (umumnya sensitif, tetapi kadang tidak begitu bagus):

```
set title '\LaTeX\ -- $ \gamma $'
```

Meletakkan label tengah-tengah baik secara horisontal maupun vertikal:

```
set label '{\LaTeX\ -- $ \gamma $}' at 0,0
```

Menentukan kedudukan (top):

```
set xlabel '[t]{\LaTeX\ -- $ \gamma $}'
```

Label lainnya – untuk label tik yang panjang:

```
set ylabel '[r]{\LaTeX\ -- $ \gamma $\rule{7mm}{0pt}'
```

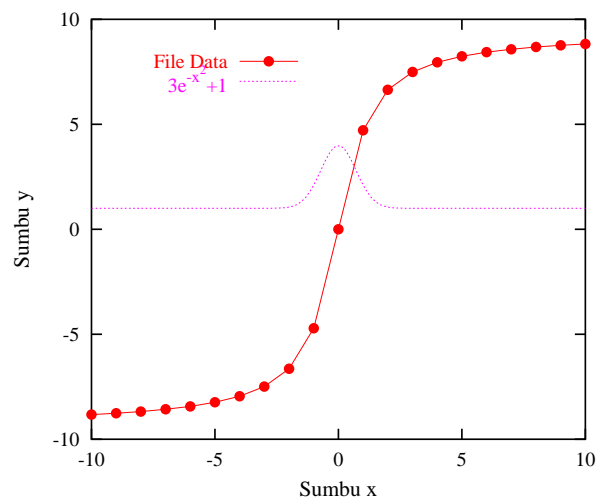
Contoh berikut ini menghasilkan grafik yang diplot ke dalam terminal L^AT_EX.

```
set terminal latex
set output "gb-8.tex"
set size 3.5/5, 3/3.
set format y "$%g$"
set format x "$%g$"
set xlabel "$x$"
set ylabel "$y$"
set key -4,8
plot [-10:10] [-10:10] "tex.dat" title "File Data" with linespoints 1 7,\
    3*exp(-x*x)+1 title "$3e^{-x^2}+1$" with lines 4
```

Jika skrip di atas dikompilasi dengan gnuplot akan dihasilkan file format $\text{T}_{\text{E}}\text{X}$ `gb-8.tex`. Untuk menambahkan file ini ke dalam dokumen $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, tinggal menambahkan satu baris tag `\input{gb-8}` pada dokumen dimana gambar akan disisipkan. Atau lebih lengkapnya adalah sebagai berikut:

```
\begin{figure}
\begin{center}
\input{gb-8.tex}
\end{center}
\end{figure}
```

Atau file ini bisa juga dijadikan file GIF. Dengan perintah `latex2html` dapat dihasilkan grafik dalam format gif dari file `gb-8.tex` ini. Hasilnya adalah sebagai berikut:



Gambar 7.1: Gambar yang dihasilkan dari dari file $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ keluaran skrip di atas

Daftar Pustaka

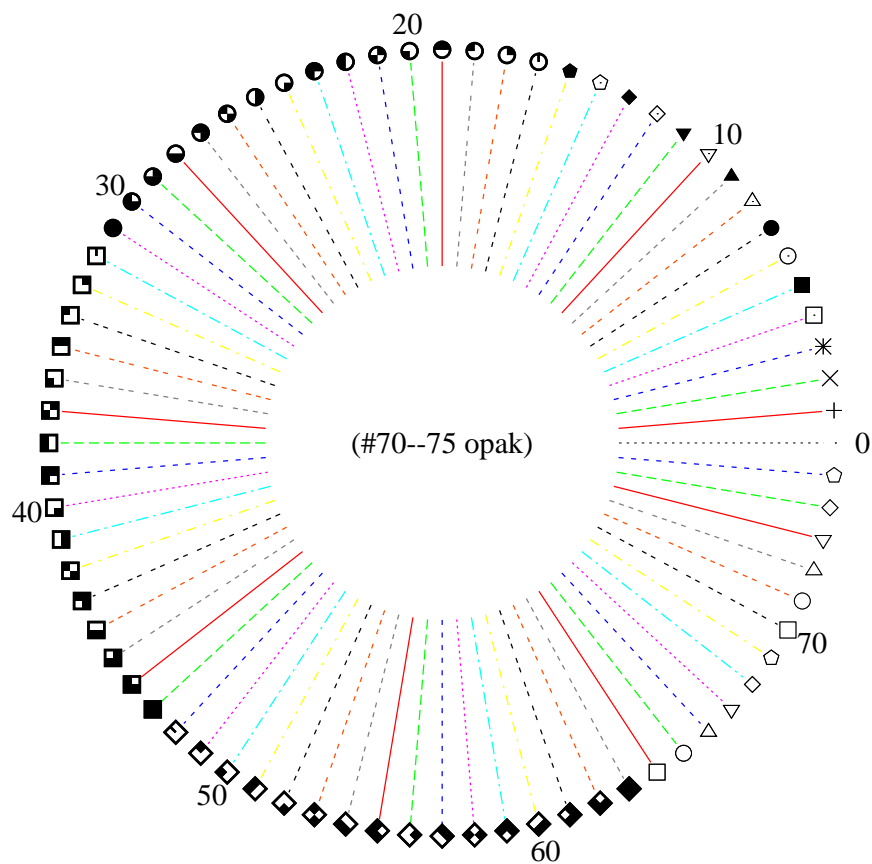
- [1] Anonymous, 2000. Gnuplot FAQ. <http://www.ucc.ie/gnuplot/gnuplot-faq.html>. Diakses tanggal 17 Januari 2001
- [2] Kotz, D., 1991. \LaTeX and the gnuplot plotting program. <http://www.gnuplot.org/>. Diakses tanggal 17 Januari 2001
- [3] Liaw, A., Crawford, D., 1994. gnuplot 3.5 User's Guide. <http://www.gnuplot.org/>. Diakses tanggal 17 Januari 2001
- [4] Sookahet, G., 1997. Premiers Pas avec Gnuplot 3.5. <http://mucha.obspm.fr/~grouss/gnuplt/gnuplot.html>. Diakses tanggal 17 Januari 2001
- [5] Williams, T., Kelley, C., 1998. Gnuplot: An interactive plotting program. <http://www.ucc.ie/gnuplot/gnuplot.html>. Diakses tanggal 17 Januari 2001

Lampiran A. Alamat beberapa situs grafik

Program	Alamat web
Gnuplot	http://www.gnuplot.org/
Xmgrace	http://plasma-gate.weizmann.ac.il/Grace/
Guppi	http://www.gnome.org:65348/guppi/
SciGraphica	http://scigraphica.sourceforge.net/
Xgfe	http://von-mises.home.ml.org/xgfe/xgfe.html
XGobi	http://www.research.att.com/areas/stat/xgobi/

Lampiran B. Data tambahan

Garis dan simbol pada terminal Postscript (skrip asli dibuat oleh Bernardo A. Mello, Brazil-ia)



Nilai parameter fungsi dari fitting ini dapat dilihat pada file fit.log berikut ini:

```
*****  
Thu Dec 28 12:31:34 2000  
  
FIT: data read from 'porosity.dat' u 2:1  
#datapoints = 30  
residuals are weighted equally (unit weight)  
  
function used for fitting: l(x)  
fitted parameters initialized with current variable values
```

```

Iteration 0
WSSR      : 4803.89          delta(WSSR)/WSSR   : 0
delta(WSSR) : 0              limit for stopping : 1e-05
lambda    : 0.732051

initial set of free parameter values
y0        = 1e-30
m         = 1e-30

After 5 iterations the fit converged.
final sum of squares of residuals : 6.93087
rel. change during last iteration : -1.62311e-10

degrees of freedom (ndf) : 28
rms of residuals      (stdfit) = sqrt(WSSR/ndf) : 0.497525
variance of residuals (reduced chisquare) = WSSR/ndf : 0.247531

Final set of parameters          Asymptotic Standard Error
=====                          =====
y0          = -35.636            +/- 2.711      (7.606%)
m           = 179.901           +/- 10.12     (5.623%)

correlation matrix of the fit parameters:
          y0      m
y0       1.000
m       -0.999  1.000

```

Data untuk menjalankan skrip diagram segitiga:

```

# *****
# isi file tern-bnd.dat
# *****
100  0  0
  0 100  0
  0  0 100
100  0  0

# *****
# isi file ter-mesh.dat
# *****
90 10  0
  0 10 90
10  0 90
10 90  0
  0 90 10
90  0 10
90 10  0

80 20  0
  0 20 80
20  0 80
20 80  0
  0 80 20
80  0 20
80 20  0

70 30  0
  0 30 70
30  0 70
30 70  0
  0 70 30

```

```
70 0 30
70 30 0

60 40 0
0 40 60
40 0 60
40 60 0
0 60 40
60 0 40
60 40 0

50 50 0
0 50 50
50 0 50
50 50 0

# *****
# Isi file ternary.dat
# *****
# Diagram fase dari Chloroform, Methanol und Ammoniak
# NH3(aq) C2H5OH CHCl3(dalam %)
96.3 0.0 3.7
87.1 8.7 4.2
79.2 15.8 5.0
67.5 27.0 5.5
58.0 34.8 7.2
50.9 40.7 8.4
45.6 45.6 8.8
40.0 50.0 10.0
33.0 54.9 12.1
24.2 60.6 15.2
13.1 65.4 21.5
6.4 64.1 29.5
0.0 50.0 50.0
```

Isi file contour.dat

```
# contour.dat
# dibuat untuk iseng...$-)
#
0 0 400
0 10 300
0 20 200
0 30 100
0 40 50
0 50 0

10 0 500
10 10 400
10 20 325
10 30 200
10 40 130
10 50 60

20 0 550
20 10 480
20 20 380
20 30 300
20 40 200
20 50 135
```

```
30 0 400
30 10 420
30 20 380
30 30 335
30 40 240
30 50 175
```

```
40 0 210
40 10 280
40 20 300
40 30 275
40 40 250
40 50 170
```

```
50 0 90
50 10 160
50 20 210
50 30 225
50 40 200
50 50 160
```

```
60 0 10
60 10 70
60 20 120
60 30 150
60 40 160
60 50 110
```

Isi file contour1.dat

```
12.5 0.0 500
20.0 7.5 500
25.0 0.0 500
4.0 0.0 400
10.0 10.0 400
20.0 18.0 400
30.0 17.0 400
32.5 10.0 400
30.0 0.0 400
0.0 10.0 300
7.5 20.0 300
20.0 30.0 300
30.0 34.0 300
39.0 30.0 300
40.0 20.0 300
38.0 10.0 300
36.0 0.0 300
0.0 20.0 200
10.0 33.0 200
20.0 40.0 200
30.0 45.0 200
40.0 45.0 200
48.0 40.0 200
52.5 30.0 200
50.0 20.0 200
45.0 10.0 200
40.0 0.0 200
0.0 34.0 100
4.0 40.0 100
12.5 50.0 100
60.0 20.0 100
```

```
56.0 10.0 100
50.0 0.0 100
60.0 50.0 95
0.0 50.0 0
60.0 0.0 0
```

Isi file tex.dat

```
#
# $Id: 3.dat,v 1.1.1.1 1998/04/15 19:16:40 lhecking Exp $
#
-20.000000 -9.125028
-19.000000 -9.109280
-18.000000 -9.091787
-17.000000 -9.072243
-16.000000 -9.050265
-15.000000 -9.025369
-14.000000 -8.996933
-13.000000 -8.964147
-12.000000 -8.925931
-11.000000 -8.880819
-10.000000 -8.826766
-9.000000 -8.760835
-8.000000 -8.678648
-7.000000 -8.573396
-6.000000 -8.433886
-5.000000 -8.240405
-4.000000 -7.954906
-3.000000 -7.494275
-2.000000 -6.642892
-1.000000 -4.712389
0.000000 0.000000
1.000000 4.712389
2.000000 6.642892
3.000000 7.494275
4.000000 7.954906
5.000000 8.240405
6.000000 8.433886
7.000000 8.573396
8.000000 8.678648
9.000000 8.760835
10.000000 8.826766
11.000000 8.880819
12.000000 8.925931
13.000000 8.964147
14.000000 8.996933
15.000000 9.025369
16.000000 9.050265
17.000000 9.072243
18.000000 9.091787
19.000000 9.109280
```