

BİLİMSEL HESAPLAMA KURSU

Mustafa Karakaplan, mustafa.karakaplan@inonu.edu.tr



Akademisyenler İçin GNU/Linux

Akademik Bilişim, Şubat 2014
Mersin Üniversitesi, Mersin

Kurs İçeriği:

1. Gün: GNU/Linux: Bash kabuğu, Linux'da tez ve makale hazırlama (LaTeX, Lyx, BibTex).
2. Gün: Python Programlamaya Giriş: Scientific, Numeric, Symbolic Python
3. Gün: Python ile bilimsel hesaplama: Veri işleme ve grafik çizme, istatistik, optimizasyon, faktoriyel tasarım, temel bileşen analizi
4. Gün: Laboratuvara Linux: Python ile Veri Alma, Gönderme, Kayıt, İşlem Kontrol, Paralel ve Seri Haberleşme (Bluetooth, USB, RS232)

Bilimsel Hesaplama Kursu
2014 - Akademik Bilişim, Mersin

Adres:

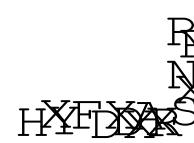
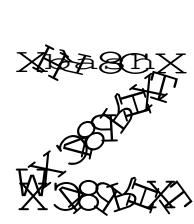
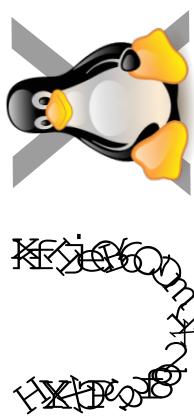
Doç. Dr. Mustafa Karakaplan
İnönü Üniversitesi, Fen-Edebiyat Fakültesi, Kimya Bölümü, Malatya
mustafa.karakaplan@inonu.edu.tr, karakaplanm@gmail.com

Bu döküman L^AT_EX ile hazırlanmıştır.

Akademik Bilişim Kursu I. Gün

GNU LINUX

Bu bölümde GNU Linux genel kullanımı hakkında bilgiler verilecektir.
Uygulamalar ve kurulumlar Ubuntu temel olarak anlatılmıştır.
Kurulumlar farklı olsa da aynı sürüm numarasına sahip uygulamaların
kullanımı ve özellikleri aynıdır.



1.1 Debian/Ubuntu

1.1.1. apt-get ile Debian/Ubuntu'ya Paket Kurmak, Kaldırmak ve Güncellemek

update	Zaman zaman paket listesini güncellemeniz gerekebilir. Depoya yeni bir paket girmiş ise böylelikle bilgisayarınız bu pakette haberdar olacaktır. \$ sudo apt-get update
install	Adını bildiğiniz bir paketi kurmak için aşağıdaki komutu kullanabilirsiniz. install komutundan sonra bir harf yazıp tab tuşuna basarsanız o harf ile başlayan paketleri görebilirsiniz. \$ sudo apt-get install paket_adi sorun olduğunda düzeltmek için; \$ sudo apt-get -f install komutunu veriniz.
upgrade	Bu komut en son paket değişikliklerini günceller. \$ sudo apt-get upgrade
remove	Bu komut ile kurulmuş paketleri kaldırabilirsiniz. Kaldırılan paketin ayar dosyaları sisteminizde duracaktır. \$ sudo apt-get remove paket_adi
purge	Kurulmuş paketleri kaldırırmakla beraber ayar dosyalarını da kaldıracaktır. \$ sudo apt-get purge paket_adi
dist-upgrade	Bu komut ile ubuntu'nun yeni sürümünü sisteminize dokunmadan kurabilirsiniz. \$ sudo apt-get dist-upgrade

1.1.2. Paket Sorgulama ve Listeleme

dpkg	Bir paket içindeki dosyaları listelemek için bu komutu kullanabilirsiniz. \$ dpkg -L gcc Bir paket hakkında içerik, tanım, kütüphane bağımlılık ve diğer bilgileri gösterebilirsiniz. \$ dpkg -s gcc
apt-cache	\$ apt-cache search algebra Bu komut ile kurulmuş veya kurulmamış paket listesi içinde verdığınız bir kelimeyi arayabilirsiniz. \$ apt-cache show gcc Bu komut ile bir paketin tanımı ve bağımlılıkları gibi bilgileri verir.



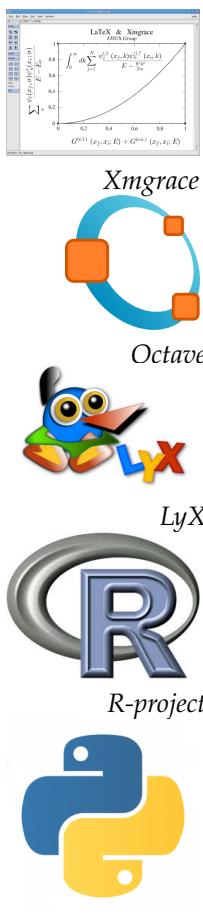
Linux dağıtımları.



Ubuntu'ya paket kurmak için en çok tercih edilen yöntem apt-get komutu ile terminalden kurulumdur. Komut ile kurulum aynı zamanda anlatım açısından da kolaydır.

Ubuntu'da kurulu paketleri veya içeriklerini sorgulama için çok tercih edilen yöntem dpkg ve apt-file komutlarıdır. apt-file komutu ön tanımlı olarak gelmez. sudo apt-get install apt-file ile kurulum yapmalısınız

1.2 Linux'da Bulunması Gereken Uygulamalar



vim	vi'in gelişmiş sürümü. \$ sudo apt-get install vim
gnuplot	Gnu çizim paketi \$ sudo apt-get install gnuplot \$ gnuplot
xmgrace	2 Boyutlu veri grafiği çizim uygulaması. Kurmak için ; \$ sudo apt-get install grace \$ xmgrace
g3data	Resim dosyalarından sayısal veri çıkarma \$ sudo apt-get install g3data \$ g3data
dia	Diagram çizim uygulaması \$ sudo apt-get install dia \$ dia
octave	matlab benzeri matematik hesaplama paketi. \$ sudo apt-get install octave \$ octave
lyx	Döküman hazırlamada LaTeX için arayüz. Kurmak için ; \$ sudo apt-get install lyx \$ lyx
R	İstatistik paketi. Kurmak için ; \$ sudo apt-get install r-base* \$ R
python	Python için bilimsel hesaplama modülleri \$ sudo apt-get install python-scipy python-scientific
arduino	Açık Kaynak Donanım programlama aracı \$ sudo apt-get install arduino \$ arduino
eagle	Elektronik şema ve baskılı devre çizim programı \$ sudo apt-get install eagle \$ eagle

2.1 UNIX Komutları

Aşağıdaki komutları çalıştmak için terminal açınız.

alias	Takma isim yaratır. Çok kullandığınız komutlara kısa takma isimler verebilirisiniz. alias ls='ls --color'
awk	Bul yerleştir, formatlı rapor oluşturma, veri tabanlarını düzenlemeye, düzenli verilerden veri madenciliği yapmada kullanılır. Aşağıdaki komut veri.dat dosyasındaki verilerin 4. ve 2. kolonlarını listeler. awk '{print \$4, \$2}' veri.dat
bzip2	Dosyaları sıkıştırma veya açma Sıkıştırma için: bzip2 dosya.txt açmak için: bunzip2 dosya.bz2
cat	Bir dosyanın içeriğini gösterir. txt dosyaları kopyalamada, dosyalara veri eklemede veya dosya birleştirmede kullanılır. cat dosya.txt cat dosya1.txt dosya2.txt > toplu.txt
cd	Klasör değiştir cd / cd .. cd /usr/local/bin
chmod	Dosya ve klasörlere erişim hakkını düzenler. chmod 744 /tmp/dosya.txt
clear	Terminal ekranını temizler clear Ctrl+l tuşuna basarak da terminal ekranını temizleyebilirsınız.
cp	Bir veya birden fazla dosyayı başka bir konuma kopyalar cp dosya.txt /tmp/ cp dosya*.txt /tmp/
date	Tarih ve zamanı değiştirmek veya görüntülemek için kullanılır date
df	Diskteki boş alanları gösterir df -h
du	O an bulunan klasör içindekilerin diskte kapladığı yeri gösterir du -h

Bu kısımda verilen UNIX komutları çoğu sisteme aynı şekilde çalışmaktadır. Solaris UNIX'de komutların kullanım şekillerinde ufak değişiklikler görülebilir. Sisteminize zarar vermemek için burada verilen komutları normal kullanıcı olarak çalıştırın.

veri.dat dosyası

1	0.1	3.4	6.5	0.3
2	0.4	1.1	0.3	0.9
3	0.7	0.4	1.4	0.6
4	0.2	0.6	0.6	1.7

Linux dizin yapısı

	bin	lib64	srv
	boot	sys	cdrom
	media	tmp	dev
	mnt	usr	etc
	opt	var	home
	proc	vmlinuz	root
	lib	sbin	lib32

	r	w	x
Sahibi	4	2	1
Grubu	4	2	1
Digerleri	4	2	1

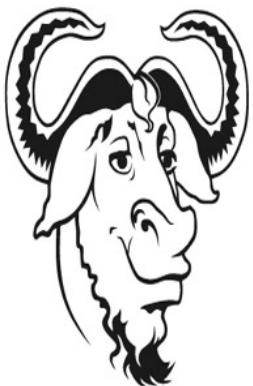
r: okuma (read)

w: yazma (write)

x: çalıştırma(execute)



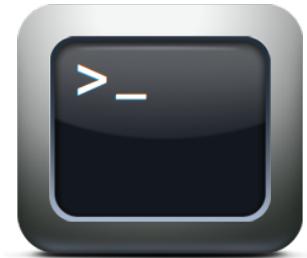
*UNIX Türevleri
RedHat, Debiab, Ubuntu
en popüler Linux
dağıtımlarıdır. Diğer
popüler UNIX türevleri ise
Solaris, OpenBSD,
NetBSD dir.*



*GNU is NOT UNIX
UNIX türevlerinde
kullanılan araçların çoğu
GNU Software Foundation
tarafından sağlanır. Bu
nedenle Linux için
GNU/Linux terimi
kullanılması salık verilir.*

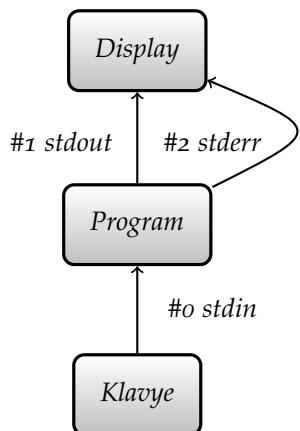
echo	ekranda mesaj görüntüler echo "Merhaba Dünya"
eject	CDROM u çıkarır eject eject -t
export	Çevresel değişkenleri düzenlemek için kullanılır export PATH=\$PATH:/home/muka/bin
find	Arama Yapma find . -exec grep aranacak find ./ -exec grep -Hn "aranacak" find /home/ -name "dene*" -print -exec rm -fr
grep	Dosya(lar) içinde verilen karakter dizisini arar. grep aranan *
history	Girilen komutların listesini verir. history !! Son verilen komutu tekrar işelevme sok !n n. ci komut çalıştır. !ev ev ile başlayan komutu yeniden çalıştır. ^eski^yeni eski ile verilen komutu yeni ile çalıştır. ^xview^xv Ctrl R Geriye doğru kullanılan bir komut ara
head	Bir dosyanın üst kısmını listeler. head -n5 dosya.txt
ispell	Text dosyada spell chek yapar. ispell dosya.txt ispell kurulu olmayabilir. Kurmak için; sudo apt-get install ispell
logname	O anki kullanılan kullanıcının login adını verir. a=\$(logname);echo \$a
logout	Giriş kabuğundan çıkmak için kullanılır. logout
ls	Dosyaların listesini ve bilgilerini gösterir ls ls -l ls -la
man	Komutlar hakkında bilgi elde etmek için kullanılır man ls

mkdir	Klasör oluşturur. Üst klasörleri oluşturmak için -p kullanabilirsiniz. mkdir klasor1 mkdir -p depo/kimya/organik	Bazı komutların root kullanıcısı yetkisi ile çalıştırılması gerekmektedir. Bu komutları çalıştmak için sudo komutu kullanılmaktadır. sudo komutu verildiğinde root kullanıcısının şifresi sorulacaktır. Terminal kapanıncaya kadar tekrar şifre sorulmayacaktır.
mount	Dosya sistemi için bağlantı noktası oluşturma. ubuntu.iso dosyasını klasor altına bağlamak için. sudo mount -o loop ubuntu.iso klasor Diskin 1. kısmında bulunan windows partition'u bir klasöre bağlamak için; sudo mount /dev/sda1 /winc	<i>sudo ile çalıştırılan komutları ne yaptığınızdan emin değilseniz denememeniz tavsiye olunur. Aksi halde sisteme zarar verebilirisiniz.</i>
mv	Dosya ve klasörlerin ismini veya yerini değiştirir. mv dosya.txt yeniad.txt	
rm	Dosya(ları) siler rm dosya.txt	
rmdir	Klasörleri siler rmdir klasor1	
sleep	Belirtilen süre kadar bekler sleep(2)	
sort	Alfabetic ve sayısal sıralama yapar. sort < dosya.txt > sirali.txt	
tail	Bir dosyanın son satırlarını gösterir. tail -n 5 dosya.txt tail -f /var/log/syslog	Terminal İkonu
tar	Dosyaları arşivleme ve arşivleri açma programı. tar -xvfz paket.tar.gz tar -cvfz yedekler.tar.gz klasor	
time	Bir programın çalışma süresini ölçer/gösterir. time programadi time sleep 2	
uname	Sistem bilgisini verir. uname -a Linux musys 3.5.0-19-generic #30-Ubuntu SMP Tue Nov 13 17:48:01 UTC 2012 x86_64 x86_64 x86_64 GNU/Linux	
whoami	Kullanıcı bilgisini gösterir.	



Terminal İkonu

2.2 (IO) Giriş Çıkış Yönlendirme



komut < dosya	Bir dosya dan komuta girdi gönderme.
komut1 < komut2	komut2 ile çalıştırılan uygulamanın çıktısını komut1 uygulamasına gönderir.
komut > dosya	komut uygulamasının standart çıktısını dosyaya yazma.
komut > /dev/null	komut uygulamasının çıktısını at.
komut >> dosya	Standart çıktıyı dosyanın sonuna ekleme.
komut 2> dosya	komut uygulamasının standart hatasını dosya'ya yazar.
komut 1> &2 dosya	Standart çıktıyı ve hatayı dosyaya yazar.
komut 2>&1 hata	standart hatayı ve çıktıyı dosyaya yazar.
komut &> dosya	Bütün çıktılar dosyaya yazılır.

2.3 Bash Programlama



Bash, Unix ve benzeri işletim sistemleri için yazılmış komut satırı kabuğu ve bu kabuğun betik dilidir. GNU Projesinin parçasıdır. GNU/Linux dağıtımlarında ön tanımlı kabuk olarak gelir. Farklı bir kabuktan bash kabuğuna geçmek için komut satırında bash komutunu vermek gereklidir. Bir kullanıcının kabuğu /etc/passwd dosyasında kayıtlıdır. Öntanımlı olmasını istediğiniz kabuğu bu dosya içerisinde belirleyebilirsiniz. Bash, sh'ın hemen hemen tüm özelliklerini ve Korn kabuğunu olan ksh ile C kabuğuna olarak bilinen csh'ın kullanıcıya özel özelliklerini bir araya getirir. Bash betiği komut satırından veilebileceği gibi .sh uzantılı bir dosyaya da yazılarak çalıştırılabilir.

Yandaki komutları prog1 şeklinde bir dosyaya yazınız. Kaydet-
tikten sonra;
chmod +x prog1
komutu ile çalıştırılabilir hale ge-
tiriniz. Çalıştmak için;
. /prog1

```

1 #!/bin/bash
clear
echo -n "Mesajınız : "
read mesaj
echo "Girilen :" $mesaj
6

```

Bash programları genelde .sh uzantılı dosyalar içine yazılır. Linuxda dosya uzantisının önemi yoktur. Uzantılar dosyaları görsel olarak ayırt edebilmek için verilir.

BASH KABUĞU

3

3.1 Döngü

For ile Döngü

```
for (( c=1; c<=5; c++ )); do ; echo "c = $c"; done
```

While ile Döngü

```
c=0; while [ $c -lt 10 ] do ; echo "c = $c"; let c++; done
```

3.2 Tarih ve Zaman

Zamanı saniye olarak yaz

```
$ date +%s  
1390141478
```

Tarihi değişken yap

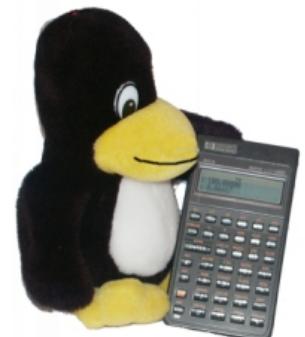
```
$ a=$(date +%Y-%m-%d); echo $a  
2014-01-19
```



3.3 SED ile İşlemler

```
ls -l | sed -e "s/[aeiou]/u/g"  
cat dene.txt| sed '/^$/d'
```

3.4 Matematik



bc ile işlem

```
$ a=20; b=1.11; c=30; echo "scale=$a;$b ^ $c" | bc  
22.89229657191140644928
```

```
$ pi=$(echo "4*a(1)" | bc -l) ; $ echo " s($pi/2)" | bc -l  
1.0000000000000000000000000000000
```

awk ile Matematik

```
$ awk -v a=22 'BEGIN{print sin(a/7)^2}'  
1.59893e-06
```

```
awk 'BEGIN{dx=0.25; for (i=1;i<10;i+=0.25) print i, sin(i)}' \\  
| xmgrace -pipe
```

Klavyeden Giriş

```
echo "2 sayı gir"; read n1 n2  
echo "$n1+$n2=$((n1+n2))"
```



3.5 avi, jpg, pdf, txt Dönüşümleri

jpg den pdf yapma

```
$ convert grafik1.jpg graf1.pdf
```

veya klasördeki jpg uzantılı dosyalardan pdf dosya yap.

```
$ convert *.jpg dosya.pdf
```

Optik Karekter Tanıma (OCR)

OCR için gerekli paketler tesseract ve tesseract-tur dir. Komut;

```
$ tesseract -l tur tarama.jpg sayfa
```

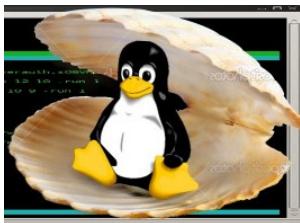


Jpg lerden video Yapma

```
mencoder "mf://*.jpg" -mf fps=8 -o video.avi -ovc lavc -lavcopts
vcodec=mjpeg
veya
vcodec=msmpeg4v2:vbitrate=800
```

3.6 Can Kurtaran Linux Komutları

```
$ minicom -D /dev/ttyUSB0 -8 -b 115200 --noinit
$ sudo dd bs=4M if=raspbian.img of=/dev/mmcblk0
$ sudo grub-install --boot-directory=/media/disk/boot /dev/sda
$ git clone git://xxx.git.sourceforge.net/gitroot/xxx
$ python -m SimpleHTTPServer # http://localhost:8000/
$ tr -cs "[[:alpha:]]" "\n" < dosya.txt
$ echo 'linux' | tr "a-z" "A-Z"
$ tr -s ' ' ' < input.txt
$ echo "34 1234 78 4" | tr [:space:] '\t'
$ tr -s '\n' ' ' < dosya.txt
$ echo "elma armut ayva" | tr " " "\n" | sort
$ cat kitap.txt | tr " " "\n" | sort | uniq -c | sort -r -n | more
```



METİN EDITÖRLERİ

4



4.1 Vi/Vim

Linux dağıtımlarında vi kurulu olarak gelir. Daha gelişkin özelliklere sahip vim sonradan kurulur. Ubuntuda kurmak için;

`sudo apt-get install vim`

komutunu vermelisiniz. Vi ye başlayabilmek için terminalden;

`vi dosya.txt`

komutunu girmelisiniz. `i` tuşuna basarak yazdıktan sonra `ESC` tuşuna bastıktan sonra `:wq` komutu vererek, kaydedip çıkış yapabilirsiniz.

Farklı bir isimde kaydetmek için `ESC` tuşuna bastıktan sonra `:w yeni.txt` şeklinde komut giriniz. Kaydetmeden çıkış yapmak için `:q!` komutunu giriniz.

4.1.1. Modlar

- | | | | |
|---|------------------------------|---|---------------------------------------|
| a | İmleçten sonra ekle. | o | Bulunulan satırın altına satır aç. |
| A | Satır sonuna ekleme yap | O | Bulunulan satırın üstüne bir satır aç |
| c | Değiştirme işlemine başla | R | Üstüne yazmaya başla |
| C | Satır sonuna kadar değiştir. | s | Bir karakter değişim |
| i | İmleçten önce yerleştir | S | Bulunulan satırı değiştir. |
| I | Satır sonuna ekleme yap. | | |

4.1.2. vi komutları

- | | | | |
|----|--|----------------|--|
| dd | imlecin bulunduğu satır silinir. | yy | İmlecin üzerinde bulunduğu satır kopyalanır. <code>p</code> ye basarak istenilen yere yapıştırılır. |
| x | imlecin üzerinde bulunduğu harf silinir. | /fig | Döküman içerisinde ileriye doğru fig kelimesini arar. aranan kelime bulunugunda imleç konumlandırılır. <code>n</code> harfine basıldığından aynı kelime aramaya devam ettirilir. |
| p | dd veya x ile kesilenler imlecin bulunduğu yerden itibaren yapıştırılır. | :1,5s/bu/deg 1 | ile 50 satırlar arasında bu kelimesini deg ile değiştirir. 1,5 yerine % konularak bütün saırlarda arama ve değiştirme yapılabilir. |
| P | dd veya x ile kesilen imlecin bir satır üstüne yapıştırılır. | | |
| cw | imlecin üzerinde bulunan kelime üzerinde değişiklik yapılır. | | |
| u | (undo) yanlışlıkla yapılan işlemler geri dönderilir. | | |

vi/vim Windows, MacOS X ve Android gibi UNIX dışındaki platformlarda da çalışabilmektedir. En popüler metin düzenleyici ünvanına sahiptir. Alştıktan sonra vazgeçilmez bir metin editörü olur. Vi veya daha gelişmiş olan Vim UNIX platformlarında metin düzenleyici olarak kullanılır.

Komut girmeden önce ESC tuşuna basmayı unutmayın.

4.1.3. İmleç hareket ettirme.

F	Bir sayfa yukarı	b	Önceki keli-)	Bir sonraki cüm-
B	Bir sayfa aşağı		menin başına	\	lenin başına
h	Sol ok	w	git		Paragraf başına
j	Sağ ok		bir sonraki keli-		git
k	Yukarı ok	e	menin başına	\	Paragraf sonuna
j	Aşağı ok	o(sıfır)	kelimenin so-		git
O	Home O	(nuna git	H	Ekranın üstüne
\$	Sona		o(sıfır) Satırın başına	M	Ekranın ortasına
			git	L	Ekranın altına
			(Cümplenin başına		git

4.1.4. Silme

dd	Bulunulan satırı sil.	X	İmleç üzerindeki karakterden önceki karakteri sil. (Geri Boşluk)
D	İmleçten satır sonuna kadar sil.	dw	kelimeyi sil
x	İmleç altındaki karakteri sil.	de	satır sonuna kadar sil
		db	satır başına kadar sil

4.1.5. Düzeltme

cw	Kelimeyi değiştir
C	Satır sonuna kadar üzereine yaz
r	tek bir harfi değiştir
s	Bir veya birden fazla karakteri bir karakter ile değiştir
S	Aktif satırı başka bir satır ile değiştir
u	Yapılan işlemi geri alır.

4.1.6. Satır numaraları

:set nu	Satır numaralarını göster
:set nonu	Satır numaralarını kapat
g	Üzerinde bulunulan satır numarasını ve dosya ismini göster.
:5	5 ci satıra git
G	son satıra git

4.1.7. Kes, Kopyala, Yapıştir

v	Mark yapmaya başla. Ok tuşları ile genişletilir. y ile kopyala, x ile kes, Pp ile yaplaştır
yy	Geçerli satırı clipboard a kopyalar.
p	Clipboard daki bilgiyi hali hazırda satırın altına kopyalar.

P	Bulunulan satırın üstüne yapıştır.
5yy	Üzerinde bulunulan satır ve sonraki 4 satırı belleğe kopyala.
:10,15w tmp.txt	10 dan 15 inci satıra kadarki bilgileri tmp.txt dosyasına yaz.
:20,\\$w tmp.txt	20. satırdan dokuman sonuna kadarki kısmı tmp.txt dosyasına yaz.
.,,10w tmp.txt	Bulunulan satırdan 10. satıra kadar olan kısmı tmp.txt dosyasına yaz.
.,,+5w tmp.txt	Bulunulan satır ve sonraki 5 satırı tmp.txt dosyasına yaz.
:r~tmp.txt	tmp.txt dosyasını aktif dosyaya ekler.

4.1.8. Arama

:set~ic	Büyük küçük harfleri dikkate alma.
:set~noic	Büyük küçük harfleri dikkate al.
/toplam	Aşağı doğru toplam kelimesini ara.
?toplam	Geriye doğru toplam kelimesini ara.
n	Son arama işlemini tekrarla.
N	ters yönde aramayı tekrarla
/^toplam	Satır başlarındaki toplam kelimesini ara.
/toplams\$	Satır sonlarındaki toplam kelimesini ara.
/[hH]oplam	Toplam veya toplam bul.

4.1.9. Değiştirme

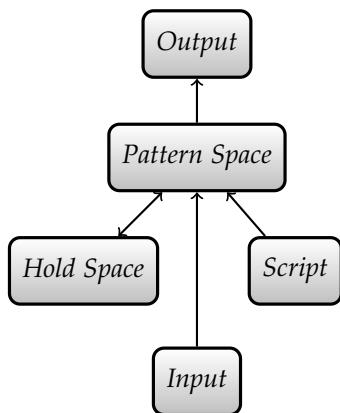
:s/ve/veya	ve leri veya olarak değiştir.
:1,5s/ve/veya	1-5 arası satırlardaki ilk ve yerine veya koy.
:1,5s/ve/veya/g	1-5 arası satırlardaki her ve yerine veya koy.
:%s/ve/veya/g	Dökümandaki bütün ve leri veya yap.
:%s/ve/veya/gc	Dökümandaki bütün ve leri veya yap ama bana sor!
xp	Birinci kelime ile ikinci kelimeyi yer değiştir.
ddp	üzerinde bulunan satır ile aşağıdaki satırı yer değiştir.

4.1.10. .vimrc dosyası

vi da yapılan ayarlamalar. .vimrc dosyasına yazılabilir.

```
{}` komutları göster
set showmode
set ic
:ab nacl Sodyum Klorür
şeklinde yazılıp kaydedildiğinde vi çalıştırıldığında yeniden ayarlama yapmaya gerek kalmayacaktır.
```

4.2 sed



sed, bir akum (stream) düzenleyicisidir. Dosya veya borudan yönlendirilmiş bilgi üzerinde metin dönüşümleri yapar ve standart çıktıya verir. sed'in en önemli özelliği bir dosyayı açmadan üzerinde işlem yapılabilmesidir. Bu nedenle çok fazla büyük metinler üzerinde bellek problemi yaşamadan değişiklik yapılmaktadır. Sed komut satırından çalıştırılır veya bir kabuk programı içerisinde konulabilir.

Milyonlarca satır içeren bir dosyadaki ondalık işaretini olarak kullanılan virgüllerin nokta yapmak için kullanılacak komut;

```
sed 's/,./g' veri.dat > yeni.dat
```

Bu komut veri.dat dosyasını tek tek okuyup virgüler yerine nokta koyarak yeni.dat dosyasına yazar.

```
aeko ları u yap
$ ls -l | sed -e "s/[aeio]/u/g"
```

```
Boş satırları sil
$ cat dene.txt| sed '/^$/d'
```

4.3 awk

awk aslında bir yorumlayıcı programlama dilidir. Girdilerden biçimlenmiş metinler oluşturmaktak kullanılır. Genel programlama yapılarını kullanarak veriler üzerinde aritmetik ve metin işlemleri gerçekleştirilebilir. Bir programın çıktısını işleyip başka bir programa girdi sağlamakta kullanabilirsiniz.

Örnek olarak birden fazla sütundan oluşan bir veri dosyasından 3. ve 1. sütünlerini alıp bir dosyaya yazmak veya bir grafik çizme programına yönlendirmek istersek, aşağıdaki şekilde bir komut yeterli olacaktır.

```
$ awk '\{ print \$3,\$1 \}' veri.dat | xmgrace -pipe
```

```
$ awk -v a=22 'BEGIN{print sin(a/7)^2}'
```

5.1 TEX / LATEX

LATEX'in çeşitli sürümleri bulunmaktadır. Ubuntu'da LATEX kurmak için; sudo apt-get install texlive komutu yeterlidir. LATEX kodu yazmak için tercih ettiğiniz bir metin düzenleyicisine ihtiyacınız olacak.

makale.tex dosyası

```
\documentclass{elsarticle}
\usepackage{lipsum}
\usepackage{natbib}
4 \begin{document}
\title{Article with \LaTeX{}}
\author{Musti Kara}
\address{Inonu University}
\begin{abstract}
9 \lipsum[1] \end{abstract}
\begin{keyword}Latex, Pdflatex
\end{keyword} \maketitle
\section{Introduction}
\lipsum[2-3] \cite{kara2012}.
14 \bibliographystyle{plain}
\bibliography{science}
\end{document}
```

science.bib dosyası

```
@article{kara2012,
title = "Introduction to LaTeX",
journal = "Latex Journal",
volume = "163",
number = "1",
pages = "54 - 60",
year = "2010",
issn = "0924-4247"
}
@book{kara2013,
title = {Article with Latex},
publisher = {Kara Yay},
year = {2012},
editor = {Musti Kara},
author = {Musti Kara}
}
```

LATEX (telafuzu "Lah-tech" veya "Lay-tech" şeklindedir) güçlü bir döküman oluşturma paketidir. Bir programlama dili olarak görülebilir. Akış kontrol komutları kullanılır ve hesaplama yapılabilir.

Yukarıdaki iki dosyayı kaydettikten sonra pdf haline dönüştürmek için;

```
pdflatex makale.tex;
bibtex makale; bibtex makale; pdflatex makale.tex
```

komutlarını giriniz. Dosyalarda bir hata yoksa makale.pdf dosyası oluşacaktır.

evince makale.pdf

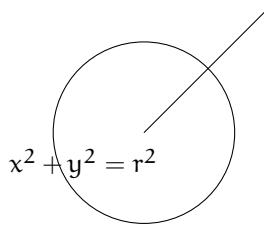
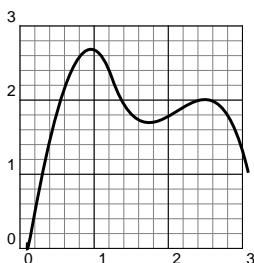
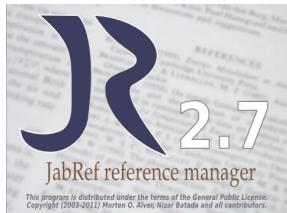
komutu ile oluşan pdf dosyasını görüntüleyebilirsiniz. latex makale.tex komutu verdığınızda pdf yerine dvi dosyası oluşacaktır.

Latex kodu içerisinde pstricks grafik kullandığınız dosyaya

```
\usepackage[pdf]{pstricks} satırını ekleyin
```

satırını ekleyin ve pdf'ye dönüştürmek için aşağıdaki şekilde derleme yapın.

```
pdflatex -shell-escape file.tex
```



5.2 LATEX Araçları

5.2.1. LyX

TEX/LATEX'i etkin kullanılmak için geliştirilmiş bir arayüzdür. Görünümü bir kelime işlem uygulamasına benzese de WYSIWYG (Ne görürsen onu alırsın) den çok WYSIWYM (Ne kasdediyorsan onu görürsün/alırsın) türü bir uygulamadır. Kurmak için; sudo apt-get install lyx komutunu girmek yeterlidir. Eğer kurulu değilse LaTeX de kurulacaktır.

5.2.2. JabRef

Java ile yazılmış bir referans düzenleyicidir. Veri tabanlarını tarayarak bibtex anahtarlarıyla birlikte makale kütüphanesi oluşturabilirsiniz. LaTeX ile kullanmak için eşsiz bir araçtır.

5.2.3. Latexdraw

PSTricks kod üretebilen bir diyagram çizime yazılımıdır. Üretilen kodu doğrudan latex kodları içeresine ekleyerek değişimli grafikler oluşturabilirsiniz.

5.3 Pgf/Tikz

PGF/TikZ geometrik, cebirsel tanımlamalar ile vektör grafik üretmeye yarayan bir dildir. PGF düşük düzeyli bir dil iken, TikZ, PGF yi kullanmaya yarayan yüksek seviyeli makrolardan oluşan bir dildir.

```
\begin{tikzpicture}\draw circle(1.5);\draw (0,0) -- (2,2); \node at (-1,-0.5){$x^2+y^2=r^2$}; \end{tikzpicture}
```

5.4 Chemfig

TikZ paketini kullanarak 2 Boyutlu molekül çizmeye yarayan bir pakettir.

```
\chemfig{*6((-O)-(-Cl)=(-CH_3)=)}=
```



```
\chemfig{A-[:30]B<[:-30]C>[:-30]~[:-30]F}
```

A-B	A — B
A=B	A == B
A~B	A ≈ B
A>B	A ▶ B
A<B	A ◀ B
A>:B	A B
A<:B	A B
A> B	A ▷ B
A< B	A ◁ B

5.5 Ekler

```
\addto\captionsenglish{\renewcommand{\contentsname}{İNDEKİLER}}
\addto\captionsamerican{\renewcommand{\contentsname}{İNDEKİLER}}
```

GRAFİK

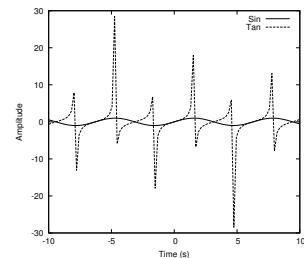
6

6.1 Gnuplot

Komut satırından komutla fonksiyon ya da verilerin iki veya üç boyutlu grafiğini çizmek için kullanılabilecek bir programıdır. Akademik makalelerde sıkça kullanılır. Birçok bilimsel hesaplama programı grafik çıktısı için gnuplot uygulamasını kullanır. Başka uygulamalardan çağırılabilir.

```
> plot sin(x) title 'Sin', tan(x) title 'Tan'  
> set xlabel "Time (s)"  
> set ylabel "Amplitude"  
4> replot  
> set grid  
> replot
```

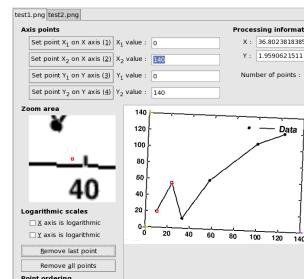
```
gnuplot> plot [-10:10] real(sin(x)**besjo(x))
```



6.2 g3data

Çok basit olmasına rağmen bazan gerek duyulan bir uygulamadır. Kağıt üzerine çizili bir görüntüden x,y noktaları elde etmek için kullanılır.

```
sudo apt-get install g3data
```



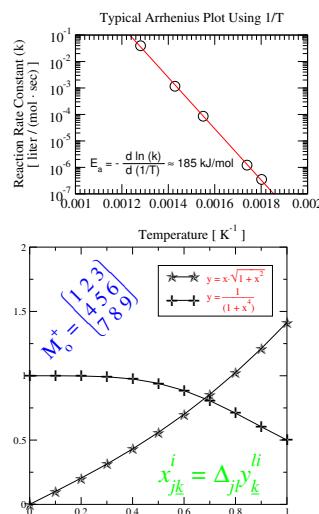
6.3 RLPlot

Qt tabanlı, yayın kalitesinde 2D ve 3D grafikler çizmek için bir program. eps ve svg çıktısı verir.

6.4 xmgrace

xmgrace 2D veri grafiği için oldukça kullanışlı bir uygulamadır. Veriler üzerinde fft, filtreleme, yumuşatma gibi işlemler gerçekleştirilebilmektedir. Ubuntu depolarında bulunmaktadır. Kurmak için;

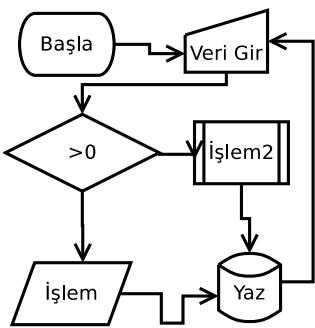
```
sudo apt-get install grace
```



6.5 GIMP

GIMP (GNU Image Manipulation Program, Türkçe: GNU Resim İşleme Programı), GNU projesi dahilinde yazılan piksel tabanlı özgür ve ücretsiz bir görüntü işleme yazılımıdır. GIMP, Adobe Photoshop ve benzeri kapalı kaynak resim işleme araçlarına eşdeğer bir işlevler bütünü sunar.

6.6 Dia



Bitmap veya Vektör grafik formatlarında çıktı verebilen diagram çizim programıdır. Algoritma, organizasyon şeması, üretim şeması gibi grafikler kolaylıkla çizilebilir.

Kurulumu;

```
sudo apt-get install dia
```

Dia bitmap grafik formatları yanında vektör grafik formatlarından eps, pdf, dxf, tex, svg, xfig yanında bir çok formatta çıktı verir. Kütüphaneleri içerisinde hazır olarak birçok alanda çizim şablonu bulunmaktadır.

6.7 XFig

Şema ve şekiller çizmek için eşsiz bir program. Başlangıçta zor gelse de alışınca vazgeçilmeyecek bir uygulama. Eps başta olmak üzere birçok grafik formatını desteklemektedir.

Kurulum ve çalıştmak için;

```
$ sudo apt-get install xfig
$ xfig
```

6.8 Inkscape

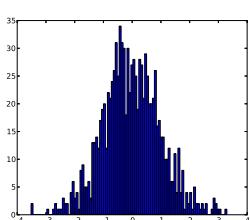
Inkscape, vektörel grafik düzenlemeye yazılımıdır. Çizimler yapabilir veya afişler hazırlayabilirsiniz. Birçok formatta girdi kabul eder ve yine birçok grafik formatında çıktı verebilir. Ölçeklenebilir grafik formatını (svg) destekler. Bitmap grafiklerden vektörel grafikler elde etmek için kullanabilirsiniz. Kurulum için aşağıdaki komutu veriniz.

```
sudo apt-get install inkscape
```

6.9 Matplotlib

Matplotlib yayın kalitesinde 2D grafikler çizmek için kullanılan bir python kütüphanesidir. pdf, eps, png ve diğer bitmap grafik formatlarında çıktılar alınabilir. matplotlib python scriptleri ile kullanılır. ipython -pylab komutu ile interaktif olarak kullanılabilir. 1000 tane gaussiyen rasgele sayının histogramunu çizmek için kullanılacak komutlar;

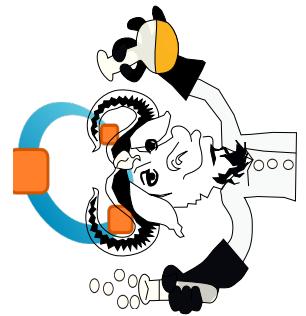
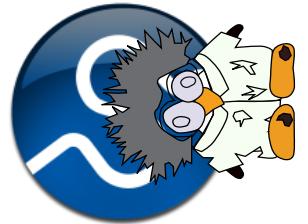
```
$ ipython --pylab
>>> x=randn(1000)
>>> hist(x,100)
```



Akademik Bilişim Kursu II. Gün

BİLİMSEL PROGRAMLAMA

Scientific Python





7.1 Python Nedir?

Python matematik paketleri (Octave, R, scilab) gibi modülleri ile birlikte gelmez. Aşağıda verilen modüller eklendiğinde python ile bilimsel hesaplama yapabilmek olanaklı hale gelir.

7.1.1. Python Başlangıç

Terminalden çalışma ve ilk kullanım aşağıdaki şekildedir.

```
$ python
>>> 2+2
4
>>> print( 'Merhaba Dünya' )
Merhaba Dünya
>>>
```

7.1.2. Birkaç İşlem

python ile temel işlemler yapmak için herhangi bir modül yüklemeye gerek yoktur.

```
>>>l=[3,5,2,6]
>>>sum(l)
16
>>> 2**10
1024
```

Python Nedir?

Python, nesne yönelimli, yorumlanabilen bir programlama dilidir. Söz dizimi girintilere dayanır. Modüler bir yapıya sahiptir. Bunun dışında diğer önemli bir özelliği de her platformda çalışmasıdır. Linux dağıtımlarında python öntanımlı bir bileşen olarak kurulu gelir.

7.2 Neden Python?

- Herşeyden önce açık kaynak kodlu özgür yazılımdır.
- Python yorumlanan bir dildir. Kod değişikliği sonrasında yeniden derlemeye gerek kalmaz.
- Python'un aynı versiyonunu kullandığınız takdirde yazdığınız kodu Windows, Mac OSX ve tüm Unix türevlerinde çalıştırabilirsiniz.
- C/C++ veya FORTRAN'da onlarca satırda yapacağınız bir işlemi tek komut ile halledebilirsiniz.
- İşletim sistemleri ve birçok uygulama betik dili olarak python kullanır. Yazılım mouse yerine komutlarla kullanmanızı sağlar.
- Bilimsel hesaplamalar yapacak bir yazılım geliştirmek için en uygun prototip geliştirme aracıdır.
- Bilimsel hesaplama için en güçlü betik programlama dilidir.
- Artıları saymakla bitmez. Tek kelime ile harika bir araçtır.

Python'un Tarihi

Geliştirilmeye 1990 yılında Guido van Rossum tarafından Amsterdam'da başlanmıştır. Adını sanılanın aksine bir yıldandan değil Guido Van Rossum'un çok sevdiği, Monty Python adlı altı kişilik bir İngiliz komedi grubunun Monty Python's Flying Circus adlı gösterisinden almıştır. Günüümüzde Python Yazılım Vakfı içerisinde toplanan gönüllülerin çabalarıyla sürdürülmektedir. Kaynak: wikipedia

7.3 Python Çalışma Ortamları

7.3.1. IPython

IPython, Python'u daha etkili kullanabilmek için zengin araçlar sunar. Etkileşimli bir arayüz sunarak Python'u daha rahat kullanmanızı sağlar.

```
$ sudo apt-get install ipython python-matplotlib
```

Çalıştırma:
matplotlib desteği ile başlatmak için.



Spyder

Python'u matlab'daki gibi bir arayüzle kullanmak isterseniz sudo apt-get install spyder

```
ipython --pylab
IPython pythondan farklı olarak In [1]: şeklinde bir imleç verir.
```

7.4 Scientific Python

Scientific Python'u scipy ile karıştırılmamalıdır. Scientific Python, Python için bilimsel araçlar içeren bir açık kaynak kütüphanedir. Aşağıdaki şekilde kurulum yapmanız gereklidir.

Scientific modülleri şöyle sıralanabilir. Clustering, Physics, Distributed Computing, FFT, Signals, Functions, Statistics, Geometry, Threading, IO, Visualization



SciPy

SciPy ayrıca popüler olan Python ile Bilimsel Programlama

Konferansının da adıdır. Konferansın web adresi <http://conference.scipy.org> dir.



$$a = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$

$$b = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

$$c = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$

$$d = \begin{bmatrix} 1 & 3 & 5 & 7 \end{bmatrix}$$

$$e = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

```
$ sudo apt-get install python-scipy python-scientific
```

Python'a Scientifiy Python modülünü yüklemek için;

```
>>> import Scientific
```

komutunu vermeniz gereklidir. Scientific alt modüllerini yüklemek için verilecek komut ise örneğin;

```
>>> from Scientific import Geometry
```

şekilde komut girmek gereklidir.

7.5 SciPy

SciPy "Sigh Pie" diye okunur ve Scientific Python demek değildir. NumPy ye bağlı olarak çalışan matematik, bilim ve mühendislik için gerekli işlemleri içeren açık kaynaklı yazılımdır. Kurulum için;

```
$ sudo apt-get install python-scipy
```

komutunu vermek gerektir. Python'a scipy modülünü yüklemek için gerekli komut;

```
>>> import scipy as sp
```

şeklindedir. Komut satırında sp. yazdıktan sonra tab tuşuna basarak scipy modülü ile gelen komutları görebilirsiniz. SciPY nin alt paketleri ise ayrıca yüklenmesi gereklidir. Örneğin;

```
>>> from scipy import linalg, optimize
```

7.6 NumPy

NumPy Python ile bilimsel hesaplama yapmak için gerekli olan temel bir pakettir. N-Boyutlu dizi nesneleri, lineer cebir, fourrier dönüşümleri ve rastgele sayı işlemlerini içerir. IPython'u çalıştırığınızda NumPy otomatik olarak yüklenirken, Python'da NumPy modülünü yüklemek için komut satırında;

```
>>> import numpy as np
```

komutunu vermeniz gereklidir.

7.7 Diğer Paketler

SYMPY Sembolik matematiksel işlemler paketi.

OPENOPT Nümerik optimizasyon paketi

PANDAS Python Data Analiz Kütüphanesi

Python ile bilimsel hesaplama için kullanılan diğer paketler;

<https://wiki.python.org/moin/NumericAndScientific>

NumPy Python ile bilimsel hesaplama yapabilmek için kullanılan temel bir pakettir. N-boyutlu dizi nesneleri, ileri düzeyde gelişmiş fonksiyonlar, C/C++ ve Fortran kodlarını entegre etmek için araçlar, lineer cebir Fourier dönüşüm ve rasgele sayı özellikleri içerir.

NumPy den yararlanabilmek için

```
>>> import numpy as np
komutunu vermeniz gereklidir.
```

```
>>> import numpy as np
>>> a = np.array([1,2,3,4,5])
>>> b = np.array([[1, 2], [3, 4], [5, 6]])
4 >>> b.shape
(3, 2)
>>> c=arange(6)
>>> d = np.arange(1, 9, 2)
>>> e = np.ones((3, 3))
```

$$\begin{aligned} a &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix} \\ b &= \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \\ c &= \begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 5 \end{bmatrix} \\ d &= \begin{bmatrix} 1 & 3 & 5 & 7 \end{bmatrix} \\ e &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \end{aligned}$$

8.1 Matrisler

```
2 >> import numpy as np
2 >> a=np.matrix([[2,3],[2,1]])
2 >> b=np.matrix([[3],[2]])
2 >> c=np.matrix([[1],[3]])
2 >> print (a*b)
7 [[12] [ 8]]
7 >> print (a*c)
7 [[11] [ 5]]
```

Matrisin transpozu
 $\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 11 \\ 5 \end{bmatrix}$

Matrisin inversi
 $\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \times \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 12 \\ 8 \end{bmatrix}$

$= 4 \times \begin{bmatrix} 3 \\ 2 \end{bmatrix}$

8.2 Matris-Array Birleştirme

```
1 >>> x = array([[1.0,2.0],[3.0,4.0]])
1 >>> y = array([[5.0,6.0],[7.0,8.0]])
1 >>> z = concatenate((x,y),axis = 0)
1 >>> z
6 array([[1., 2.],
[3., 4.],[5., 6.],[7., 8.]])
6 >>> z =concatenate
6 ((x,y),axis = 1)
6 >>> z
11 array([[ 1.,  2.,  5.,  6.],
[ 3.,  4.,  7.,  8.]])
```

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100

$$\begin{aligned} 3x - 2y + z &= 4 \\ x + y - 2z &= 7 \\ 2x - y + 3z &= 1 \end{aligned}$$

$$\left| \begin{array}{ccc|c} 3 & -2 & 1 & |x| \\ 1 & 1 & -2 & |y| \\ 2 & -1 & 3 & |z| \end{array} \right|$$

3

```
>> A = np.array([[3,-2,1],[1,1,-2],[2,-1,3]])
>> b = np.array([[4],[7],[1]])
>> x = np.linalg.solve(A, b)
>> x
array([[ 3.],
       [ 2.],
       [-1.]])
```

$$A = \begin{bmatrix} 3 & 0 & 1 \\ -4 & 1 & 2 \\ -6 & 0 & -2 \end{bmatrix}$$

$$Av = \lambda v$$

$$\det(A - \lambda I) = 0$$

8.4 Özdeğer ve Özvektör

Bir λ sayısının A matrisinin özdeğeri olması

$$Ax = \lambda x$$

koşulunu sağlayan $x \neq 0$ olacak şekilde bir x vektörünün var olmasıdır.

```
>> A=np.matrix([[3,0,1],[-4,1,2],[-6,0,-2]])
>> np.linalg.eig(A)
array([1.0, 1.0, 0.0])
matrix([[ 0.0, 0.0, -0.095],
       [ 1.0, -1.0, -0.095],
       [ 0.0, 0.0, 0.286]]))
```

8.5 Fortran Modül Ekleme



<pre>C File topla.f subroutine topla (n) integer n,t t=0 do 100 i=0, n t=t+i print *, i,t 100 continue end</pre>	<pre>>>> !f2py -c -m topla topla.f >>> import topla >>> topla.topla(4) 0 0 1 1 2 3 3 6 4 10</pre>
---	---

8.6 Ekler

```
data = np.genfromtxt('veri.dat', delimiter = ',')
plot(data[:,0], data[:,1])
```

PYTHON İLE GRAFİK

9

9.1 Matplotlib

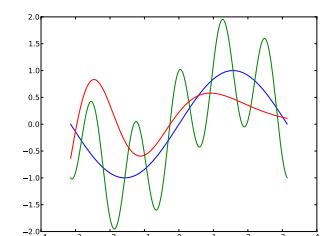
Mathplotlib paketini daha önce kurmuştuk. Eğer bu paket kurulu değilse;

```
$ sudo apt-get install python-matplotlib
```

```
1 $ ipython --pylab  
2 >>> import matplotlib.pyplot as plt
```

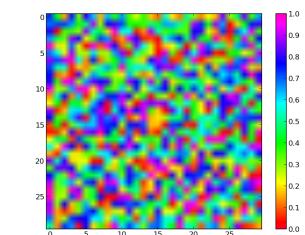
9.1.1. Fonksiyon Grafiği

```
In [1]: x=arange(-pi,pi,0.01); y=sin(x)  
In [2]: plot(x,y); plot(x,sin(x)+cos(x*5))  
3 In [3]: from scipy.special import jn  
In [4]: plot(x,jn(x,2))
```



9.1.2. Görüntü Grafiği

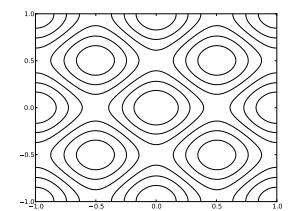
```
1 In [1]: image=np.random.rand(30,30)  
2 In [2]: plt.imshow(image,cmap=plt.cm.gist_rainbow)  
3 In [3]: plt.colorbar(); plt.show()
```



9.1.3. Contour Plot

Rastrigin Fonksiyonu Contour Grafik

```
import numpy as np  
2 import matplotlib.pyplot as plt  
plt.figure()  
xlist = np.linspace(-1.0,1.0,100); ylist = np.linspace(-1.0,1.0,100)  
X,Y = np.meshgrid(xlist, ylist)  
Z = 20+(X**2-10*np.cos(2*np.pi*X)) + (Y**2-10*np.cos(2*np.pi*Y))  
7 plt.contour(X, Y, Z, colors = 'k', linestyles = 'solid')  
plt.show()
```

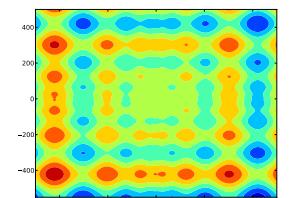


Schwefel Fonksiyonu Contour Grafik

```
import numpy as np  
2 import matplotlib.pyplot as plt  
plt.figure()  
xlist = np.linspace(-500, 500.0, 100)  
ylist = np.linspace(-550, 500.0, 100)  
X,Y = np.meshgrid(xlist, ylist)  
7 Z = -X*np.sin(np.sqrt(np.abs(X)))-Y*np.sin(np.sqrt(np.abs(Y)))  
    +418.9829*2  
plt.contourf(X, Y, Z)  
plt.show()
```

$$A_n + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)]$$

$$A = 10, x_i \in [-5.12, 5.12]$$

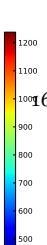
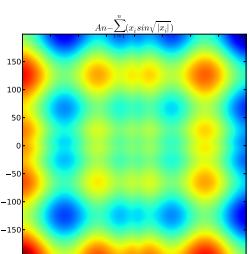
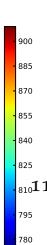
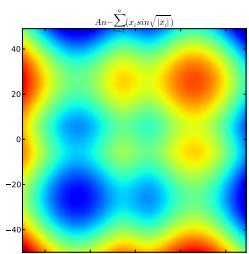
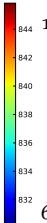
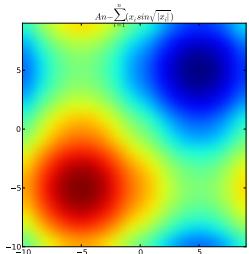


$$A_n - \sum_{i=1}^n (x_i \sin \sqrt{|x_i|})$$

$$A = 418.9829, x_i \in [-500, 500]$$

9.2 Hareketli Grafik

Hareketli grafik oluşturmak ve bunu avi olarak kaydetmek için aşağıdaki kod verilmiştir. 500 kare resim oluşturulup output.avi dosyası oluşturulmaktadır. Öncelikle sudo apt-get install mencoder ile gerekli mencoder paketini kurmanız gereklidir. Oluşan video dosyasını yüklü olan herhangi bir video oynatıcısı ile açabilirsiniz.



```
import subprocess; import numpy as np
import matplotlib.pyplot as plt

plt.title(r'$An = \sum_{i=1}^n (x_i \sin \sqrt{|x_i|})$', size=14)
aralik = np.arange(1,500,1)
for i in aralik:
    x, y = np.mgrid[-i:i, -i:i]
    z=-x*np.sin(np.sqrt(np.abs(x)))-y*np.sin(np.sqrt(np.abs(y)))
    +418.9829*2
    plt.imshow(z, origin='lower', interpolation='bicubic',
               extent=(x.min(), x.max(), y.min(), y.max()))
    plt.colorbar()
    filename = str('%03d' % i) + '.png'
    plt.savefig(filename, dpi=100)
    plt.show();plt.clf()

#png dosyalardan avi yap
command = ('mencoder mf://*.png -mf type=png:w=800:h=600:fps=25',
           '-ovc lavc -lavcopts vcodec=mpeg4 -oac copy',
           '-o output.avi')
subprocess.check_call(command)
```

9.3 xmgrace ile Haberleşme

```
from pygrace import grace
pg = grace();
x=arange(0,2*pi,0.1);
pg.plot(x,sin(x));pg.xlabel('Absis')
```

9.4 Ekler

```
plt.xlim(5,18);plt.ylim(-5,5)
plt.xticks([1,2],['38','39']); plt.yticks([2,4],['39','40'])
plt.text(30,30,"Text",ha="center",family='times',size=10)
x=np.random.randn(1000); plt.hist(x,bins=30)

b,g,r,c,m,y,k,w; Solid - Dashed -- Dash-dot - . Dotted :
Point . Pixel Circle o Square s Diamond D Thin Diamond d Cross +
Plus + Star * Hexagon H Alt. Hexagon h Pentagon p Triangles
```

Python ile sembolik matematik için gerekli paketi aşağıdaki gibi kurun.
`sudo apt-get install python-sympy`

```
>>> import sympy as sm
>>> x = sm.Symbol('x') ; y = sm.Symbol('y')
>>> sm.expand((x-2)*(x+3))
x**2 + x - 6
5   >>> f=sm.expand((x-2)*(x+3)*(x-y))
    >>> f
x**3 - x**2*y + x**2 - x*y - 6*x + 6*y
>>> f.subs({x:1,y:3})
    8
10  >>> sm.expand(sm.cos(x+y), trig=True)
    -sin(x)*sin(y) + cos(x)*cos(y)
    >>> sm.simplify((2*x**2+(x*y)**2)/x**2)
    y**2 + 2
```

$$\begin{aligned}\cos(2\theta) &= \cos^2 \theta - \sin^2 \theta \\(x+y)^2 &= x^2 + 2xy + y^2 \\(x+y)(x-y) &= x^2 - y^2 \\(x+2)(2x-5) &= 2x^2 - x - 10 \\(x+y)^6 &= 1x^6 + 6x^5y \\&\quad + 15x^4y^2 + 20x^3y^3 \\&\quad + 15x^2y^4 + 6xy^5 + 1y^6 \\ \sin(x+y) &= \\ \sin x \cos y + \cos x \sin y &= \\ \cos(x+y) &= \\ \cos x \cos y - \sin x \sin y &= \\ \sin(x-y) &= \\ \sin x \cos y - \cos x \sin y &= \\ \cos(x-y) &= \\ \cos x \cos y + \sin x \sin y &=\end{aligned}$$

10.1 Limit

```
2   >>> sm.limit(1/x, x, oo)
0
>>> sm.limit(sm.sin(x)/x, x, 0)
1
3   >>> sm.limit((3*(x+y)**2-3*x**2)/y,y,0)
    >>> 6*x
```

$$\begin{aligned}\lim_{x \rightarrow \infty} \exp(-x) &= 0 \\ \lim_{x \rightarrow 0} \frac{\sin x}{x} &= 1 \\ \lim_{x \rightarrow 0} \frac{1 - \cos x}{x} &= 0 \\ \lim_{x \rightarrow \infty} x \sin\left(\frac{c}{x}\right) &= c\end{aligned}$$

10.2 Türev

```
4   >>> sm.diff(sm.sin(x), x)
cos(x)
>>> sm.diff(3*x**2)
    >>> 6*x
```

$$\begin{aligned}\frac{d}{dx} (\ln|x|) &= \frac{|x|}{x^2} \\ \frac{d}{dx} (\ln x) &= \frac{1}{x}, \quad x \neq 0 \\ \frac{d}{dx} (\log_c x) &= \frac{1}{x \ln c}\end{aligned}$$

10.3 İntegral

```
1   >>> sm.integrate(3*x**2+2*x-4)
x**3 + x**2 - 4*x
>>> import math as m
>>> sm.integrate(sm.cos(x), (x, -m.pi/2, m.pi/2))
2
```

$$\begin{aligned}\int_0^\infty e^{-x} dx &\\ \int_0^{+\infty} x^n e^{-x} dx &= n! \\ \int \cos \theta d\theta &= \sin \theta \\ \int_0^R \frac{2x dx}{1+x^2} &= \log(1+R^2)\end{aligned}$$

10.4 Seriler

$$\sum_{n=1}^{\infty} \frac{1}{2^n} = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots$$

```
>>> sm.series(sm.sin(x), x)
x - x**3/6 + x**5/120 + O(x**6)
```

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

10.5 Denklem Sistemleri Çözümü

```
>>> sm.solve([x + 5*y - 2, -3*x + 6*y - 15], [x, y])
{y: 1, x: -3}
```

10.6 Matrisler

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} =$$

$$\frac{1}{\det(\mathbf{A})} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} =$$

$$\frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc$$

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0$$

```

3      >>> A = sm.Matrix([[1,x], [y,1]])
4      >>> A
5      [1, x]
6      [y, 1]
7      >>> A**2
8      [1 + x*y,      2*x]
9      [      2*y, 1 + x*y]

10     >>> sm.Matrix.det(A)
11     -x*y + 1

12     >>> sm.Matrix.eigenvals(A)
13     {sqrt(x*y) + 1: 1, -sqrt(x*y) + 1: 1}

14     >>> sm.Matrix.eigenvects(A)
15     [(-sqrt(x*y) + 1, 1, [[-x/sqrt(x*y)][1]]),
16      (sqrt(x*y) + 1, 1, [[x/sqrt(x*y)][1]])]

17     >>> A.subs({x:3,y:7})
18     [1, 3]
19     [7, 1]
```

10.7 LaTeX Denklem

$$\int \frac{x^2 + e^{x^2} \sin(x)}{(x+1)(x+2)} dx$$

```

print(sm.latex(sm.Integral(x**2+sm.sin(x)*sm.exp(x**2),x),mode='
    equation', itex=True))
3 $$\int x^{2 } + e^{x^{2 }} \sin \left( x \right) dx
```

```

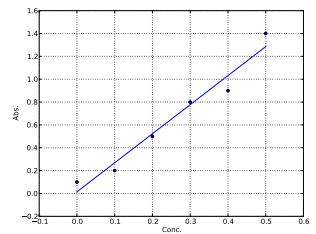
>>> a = 1/(x+2)*(x+1)
>>> sm.latex(a)
' \frac{1}{\left( x + 2 \right) \left( x + 1 \right) }'
```

11.1 Lineer-Polinom

```

1 from pylab import *
2 x=[0.0, 0.1, 0.2, 0.3, 0.4, 0.5]
y=[0.1, 0.2, 0.5, 0.8, 0.9, 1.4]
(m,b)=polyfit(x,y,1); yy=polyval([m,b],x)
plot(x,yy); scatter(x,y);grid(True)
xlabel('Conc.'); ylabel('Abs.');
show()

```



3. satırda `sonuc=polyfit(x,y,2); yy=polyval(sonuc,x)` kodunu kullanarak 2. derece polinoma uydurmayı deneyiniz.

11.2 Python

```

from scipy import optimize
def f(x):
    return x[0]**2 + (x[1]-2)**2
print optimize.fmin(f,[0,0])

```

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

11.3 OpenOPT

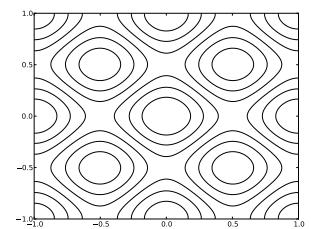
```

1 from openopt import GLP
from numpy import *

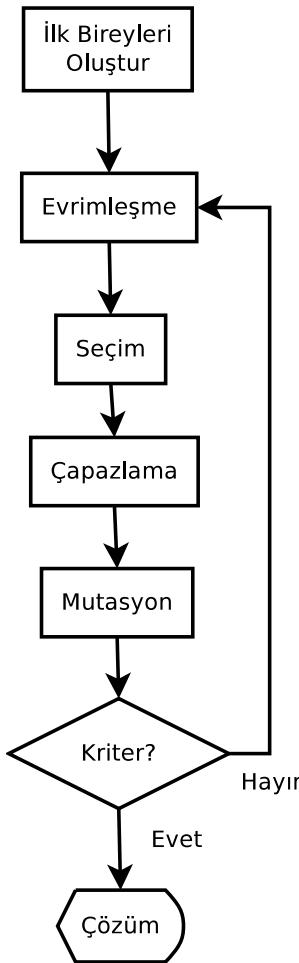
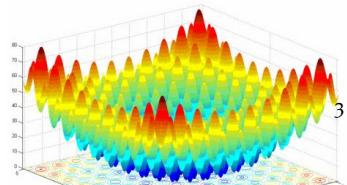
#Rastrigin function
f = lambda x: (x*x - 10*cos(2*pi*x) + 10).sum()
6 p = GLP(f, lb = -ones(2)*5.12, ub = ones(2)*5.12,
          maxIter = 1e3, maxFunEvals = 1e5,
          maxTime = 100, maxCPUTime = 100)

r = p.solve('de', plot=0)
11 x_opt, f_opt = r.xf, r.ff
print x_opt

```



11.4 PyEvolve



```

from pyevolve import G1DList, GSsimpleGA
from pyevolve import Initializators, Mutators, Consts
import math

# This is the Rastrigin Function, a deception function
def rastrigin(xlist):
    n = len(xlist)
    total = 0
    for i in range(n):
        total += xlist[i]**2 - 10*math.cos(2*math.pi*xlist[i])
    return (10*n) + total

# Genome instance
genome = G1DList.G1DList(2)
genome.setParams(rangemin=-5.2, rangemax=5.30)
genome.initializer.set(Initializators.G1DListInitializerReal)
genome.mutator.set(Mutators.G1DListMutatorRealGaussian)

# The evaluator function (objective function)
genome.evaluator.set(rastrigin)

# Genetic Algorithm Instance
ga = GSsimpleGA.GSsimpleGA(genome)
ga.minimax = Consts.minimaxType["minimize"]
ga.setGenerations(800)
ga.setMutationRate(0.05)

# Create DB Adapter and set as adapter
#sqlite_adapter = DBAdapters.DBSQLite(identify="rastrigin")
#ga.setDBAdapter(sqlite_adapter)

# Do the evolution, with stats dump
# frequency of 10 generations
ga.evolve(freq_stats=50)

# Best individual
best = ga.bestIndividual()
print "\nBest individual score: %.2f" % (best.getRawScore(),)
print best
  
```

12.1 Temel İstatistik

```

import numpy as np

5   >>> x=[ 4, 6, 3, 9, 8, 4, 7]
    >>> y=[12,14,17,11,17,12,15]
    >>> np.mean(x)
    5.8571428571428568
    >>> np.median(x)
    6.0
    >>> np.std(x)
    2.0995626366712958
    >>> np.var(x)
    4.4081632653061229
    >>> np.cov(x,y)
    array([[5.14285714, -0.5],
    15      [-0.5,  6.]])

```

```

import numpy as np
x = np.linspace(-3, 3, 100)
h = plt.plot(x, norm().pdf(x))

```

```

2   >>> import scipy as sc
    >>> sc.stats.norm(0,1).pdf(0)
    0.3989422804014327
    >>> sc.stats.norm(0,1).cdf(0)
    0.5

```

```

x=np.linspace(0, 10, 100)
h=plot(x,
        sc.stats.norm.pdf(x,5,2))
h=plot(x,
        sc.stats.norm.pdf(x,5,1))
5

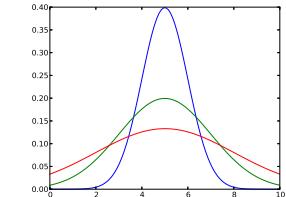
```

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$$

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i$$

$$pdf(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

$$f(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$



12.2 Testler

```

>>> x=[3,2,4,4,5,4,2,9];sc.stats.skewtest(x)
(2.1649, 0.0304)

>>> =x=[4,6,3,9,6]
>>> print 't-st= %6.3f pval = %6.4f'
      % sc.stats.ttest_1samp(x, 0.05)
t-st=  5.391 pval = 0.0057

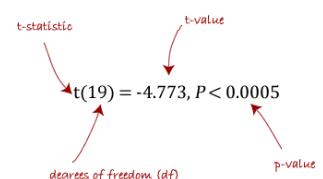
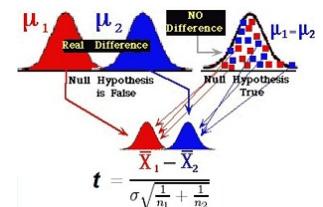
>>> print 'KS-statistic D = %6.3f pvalue = %6.4f'
      % stats.kstest(x, 't', (10,))
KS-statistic D =  0.993 pval = 0.0000

>>> x1=[6,4,5,5,3];x2=[7,6,9,4,5]
>>> sc.stats.ttest_ind(x1, x2)
(array(-1.600), 0.1483)

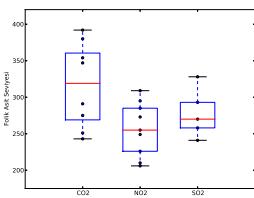
>>> sc.stats.ks_2samp(x1,x2)
(0.4000, 0.6974)

```

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$



12.3 ANOVA



```

from scipy import stats; from numpy import hstack, array
from pylab import plot, show, savefig, xlim, figure, repeat,\n    hold, ylim, legend, boxplot, scatter, setp, axes
from statsmodels.stats.multicomp import (pairwise_tukeyhsd,\n    MultiComparison)
5 C02 = array([243,251,275,291,347,354,380,392])
N02 = array([206,210,226,249,255,273,285,295,309])
S02 = array([241,258,270,293,328])
[f_val, p_val] = stats.f_oneway(C02, N02, S02)
print "F =", f_val ; print "P =", p_val
10 marks = hstack([C02, N02, S02])
group=hstack([[ 'CO2']*len(C02),[ 'NO2']*len(N02),[ 'SO2']*len(S02)])
res2 = pairwise_tukeyhsd(marks,group,alpha=0.10)
mod = MultiComparison(marks, group)
print mod.tukeyhsd(); print mod.tukeyhsd()[0]; print res2; print
    res2[0]
15 output = (f_val, p_val,S02.mean(), S02.std(),
            N02.mean(), N02.std(), C02.mean(),C02.std())
print 'F(2,27) =%6.2f , p =%6.2f. ' % (f_val, p_val)
print 'CO2 (M =%6.2f,SD =%6.2f)' % (N02.mean(), N02.std())
print 'NO2 (M =%6.2f, SD =%6.2f)' % (C02.mean(),C02.std())
20 print 'SO2 (M =%6.2f, SD =%6.2f)' % (S02.mean(), S02.std())
fig = figure(); ax = axes(); hold(True)
boxplot(C02,positions=[1],widths=0.3);
boxplot(N02,positions=[2],widths=0.3)
boxplot(S02,positions=[3],widths=0.3);
25 scatter(repeat(1,len(C02)),C02)
scatter(repeat(2,len(N02)),N02); scatter(repeat(3,len(S02)),S02)

xlim(0.5,3.5); ylim(175,420)
ax.set_xticklabels(['CO2', 'NO2', 'SO2']); ax.set_xticks([1, 2, 3])
30 ax.set_ylabel('Folik Asit Seviyesi')
show()

```

12.4 Ekler

```

random.seed(5); random.random(); random.randrange(1,10+1);
randint(1,6); x = arange(10); shuffle(x);x; shuffle(x);x
x = arange(10); permutation(x);permutation(x);permutation(x)
random.gauss(50,30);random.normalvariate(50,30)

```

13.1 PyDOE

PyDOE python paketleri arasında bulunmaz. setupool ile yüklemek gerekiyor.

```
# sudo apt-get install python-setuptools
```

ile python setupool araçını yükledikten sonra

```
# sudo easy_install --upgrade pyDOE
```

ile PyDOE paketini yükleyebilirsiniz.

```
>>> from pyDOE import *
```

13.1.1. Full Faktoriyel

2 ve 3 seviyeli 2 parametre

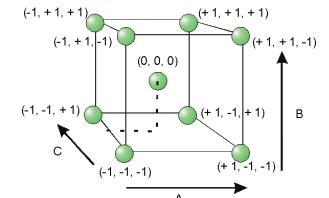
```
>>> fullfact([2, 3])
array([[ 0.,  0.],
       [ 1.,  0.],
       [ 0.,  1.],
       [ 1.,  1.],
       [ 0.,  2.],
       [ 1.,  2.]])
```

2,3 ve 1 seviyeli 3 parametre

```
>>> fullfact([2,3,1])
array([[ 0.,  0.,  0.],
       [ 1.,  0.,  0.],
       [ 0.,  1.,  0.],
       [ 1.,  1.,  0.],
       [ 0.,  2.,  0.],
       [ 1.,  2.,  0.]])
```

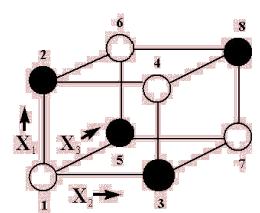
13.1.2. 2-Level Full-Factorial (ff2n)

```
>>> ff2n(3)
array([[-1., -1., -1.],
       [ 1., -1., -1.],
       [-1.,  1., -1.],
       [ 1.,  1., -1.],
       [-1., -1.,  1.],
       [ 1., -1.,  1.],
       [-1.,  1.,  1.],
       [ 1.,  1.,  1.]])
```



13.1.3. 2-Level Fractional-Factorial (fracfact)

```
>>> fracfact('a b ab')
array([[-1., -1.,  1.],
       [ 1., -1., -1.],
       [-1.,  1., -1.],
       [ 1.,  1.,  1.]])
```



Run	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈
1	+	+	+	+	+	+	+	+
2	-	+	-	+	+	+	-	+
3	-	-	+	-	+	+	+	+
4	+	-	-	+	-	+	+	+
5	-	+	-	-	+	-	+	-
6	-	-	+	-	-	+	-	+
7	-	-	-	+	-	-	+	+
8	-	-	-	-	-	-	-	-

13.1.4. Plackett-Burman (pbdesign)

```
>>> pbdesign(3)
array([[-1., -1., 1.],
       [1., -1., -1.],
       [-1., 1., -1.],
       [1., 1., 1.]])
```

13.2 Uygulama

	X ₁	X ₂	X ₃
1	-1	-1	-1
2	+1	-1	-1
3	-1	+1	-1
4	+1	+1	-1
5	-1	-1	+1
6	+1	-1	+1
7	-1	+1	+1
8	+1	+1	+1

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$$

için çözüm aşağıdadır.

	(-1)	(0)	(+1)	Birim
Hız	16	20	24	rpm
Nem	0.1	0.3	0.5	mg/L
pH	3.0	3.5	4.0	



No	Hız	Nem	pH	Verim
1	16	0.1	3	62.4
2	24	0.1	3	90.6
3	16	0.5	3	63.5
4	24	0.5	3	91.7
5	16	0.1	4	61.8
6	24	0.1	4	88.3
7	16	0.5	4	64.9
8	24	0.5	4	90.2

Yukarıdaki örnek uygulamamızda Hız parametresi katsayı olarak en büyük değere sahip olduğundan bu parametrenin sonuca etkisi en fazladır. Diğer parametrelerden Nem 0.9 katsayı ile 2. önemli etkendir.

Parametreler arasında etkileşim birinci dereceden etkişelim olup olmadığını anlamak için aşağıdaki denklem sistemini oluşturup çözümü siz bulunuz.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_{12} X_1 X_2 + \beta_{13} X_1 X_3 + \beta_{23} X_2 X_3 + \beta_{123} X_1 X_2 X_3 + \epsilon$$

ffornek.py dosyası

```
import numpy as np
A=np.zeros([8,4])
A[:,0]=1
4 A[:,1:4]=ff2n(3)
B=np.zeros([8, 3])
B[:,0]=20+A[:,0]*4
B[:,1]=0.3+A[:,1]*0.2
B[:,2]=3.5+A[:,2]*0.5
9 b=np.array([[62.4],[90.6],
[63.5],[91.7],[61.8],
[88.3],[64.9],[90.2]])
x,res,rnk,s=np.linalg.lstsq(A,b)
print x
```

Çıktı

```
[[ 76.675]
 [ 13.525]
 [ 0.9 ]
 [ -0.375]]
```

$$Y = 76.7 + 13.5\text{Hz} + 0.9\text{Nem} - 0.37\text{pH}$$

matplotlib paketi PCA modülü içermektedir.

```
1 from matplotlib.mlab import PCA
data = array(randint(10,size=(10,2)))
results = PCA(data)
```

14.1 sklearn.decomposition.PCA

```
import numpy as np
2 from sklearn.decomposition import PCA
X = np.array([[-1, -1], [-2, -1], [-3, -2], [1, 1], [2, 1], [3, 2]])
pca = PCA(n_components=2)
pca.fit(X)
print(pca.explained_variance_ratio_)
```

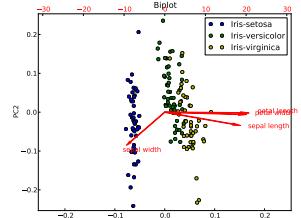
14.2 orngPCA

sudo easy_install --upgrade orange

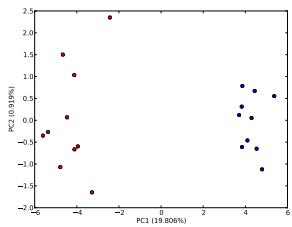
```
import orange, orngPCA

data = orange.ExampleTable("iris.tab")
4 attributes = ['sepal length', 'sepal width', 'petal length', 'petal
    width']
pca = PCA(data, standardize = True, attributes = attributes)

pca(data)
pca.biplot()
9
print pca
```



14.3 MDP (Modular Toolkit for Data Processing) ve PCA



```

import mdp
import numpy as np
import matplotlib.pyplot as plt

5 #Create sample data
var1 = np.random.normal(loc=0., scale=0.5, size=(10,5))
var2 = np.random.normal(loc=4., scale=1., size=(10,5))
var = np.concatenate((var1,var2), axis=0)

10 #Create the PCA node and train it
pcan = mdp.nodes.PCANode(output_dim=3)
pcar = pcan.execute(var)

#Graph the results
15 fig = plt.figure()
ax = fig.add_subplot(111)
ax.plot(pcar[:10,0], pcar[:10,1], 'bo')
ax.plot(pcar[10:,0], pcar[10:,1], 'ro')

20 #Show variance accounted for
ax.set_xlabel('PC1 (%.3f%%)' % (pcan.d[0]))
ax.set_ylabel('PC2 (%.3f%%)' % (pcan.d[1]))

plt.show()

```

Görüntü işleme ve kameradan görüntü almak için gerekli paket

```
sudo apt-get install python-imaging python-pygame
```

```

1 #File: cam.py
import pygame.camera
pygame.camera.init()
cam = pygame.camera.Camera(pygame.camera.list_cameras()[0])
cam.start()
6 img = cam.get_image()
import pygame.image
pygame.image.save(img, "foto.jpg")
pygame.camera.quit()
```



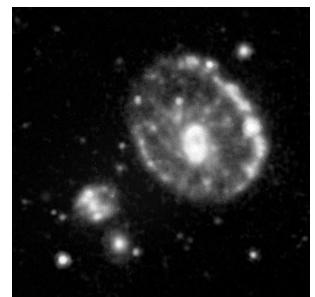
Renkli

```

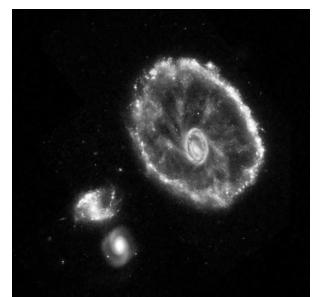
1 #File: rgb.py
import numpy
import Image

image=Image.open('Cartwheel.jpg')
6
xsize, ysize=image.size
r, g, b=image.split()
r.show() ; g.show() ; b.show()
rdata=r.getdata(); gdata=g.getdata() ; bdata=b.getdata()
11
npr=numpy.reshape(rdata, (ysize, xsize))
npg=numpy.reshape(gdata, (ysize, xsize))
npb=numpy.reshape(bdata, (ysize, xsize))

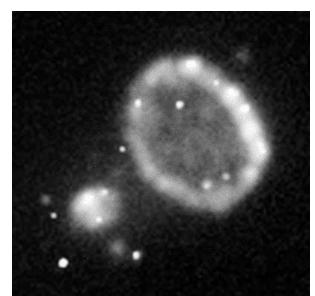
16 rgbArray = np.zeros((ysize,xsize,3), 'uint8')
rgbArray[..., 0] = npr/2.0
rgbArray[..., 1] = npg/3.0
rgbArray[..., 2] = npg/4.0
img = Image.fromarray(rgbArray)
21 img.show()
img.save('yeni.jpg')
```



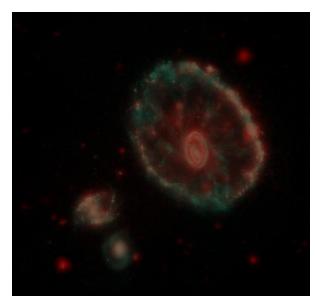
Red



Green



Blue



R,G,B kombine

15.1 Bazı komutlar

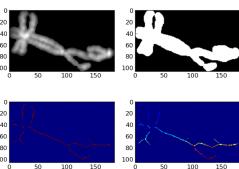
```
>>> r,g,b=image.getpixel((10,10)) # 10,10 daki rengi al
>>> image.putpixel((10,10),(255,0,0)) #kırmızı yap
```

15.2 Mahotas

Mahotas Python için görüntü işleme ve bilgisayar görüntüleme için bir seri fonksiyonlar içeren bir pakettir. Biyogörüntü Sistemleri için geliştirilmiş olmasına rağmen bütün alanlarda kullanılmaktadır.

Kurulum için;

```
$ sudo easy_install --upgrade mahotas
$ sudo easy_install --upgrade pymorph
```



```

1 import pylab
2 import pymorph
3 import mahotas
4 from scipy import ndimage
5
6 image = mahotas.imread('stars_clusters.jpg')
7 pylab.imshow(image)
8 pylab.gray()
9 pylab.show()
10
11 print image.shape
12 print image.dtype
13 print image.max()
14 print image.min()
15
16 pylab.imshow(image // 2)
17 pylab.show()
18
19 T = mahotas.thresholding.otsu(image)
20 pylab.imshow(image > T)
21 pylab.show()
22
23 print "Gaussian Filter"
24 imagef = ndimage.gaussian_filter(image, 1)
25 T = mahotas.thresholding.otsu(imagef)
26 pylab.imshow(imagef > T)
27 pylab.show()
28
29 labeled,nr_objects = ndimage.label(imagef > T)
30 print 'Yıldız Sayısı=%d' % nr_objects
31 pylab.imshow(labeled)
32 pylab.jet()
33 pylab.show()
```

16.1 Hiyerarşik Clustering

```

>>> from cluster import *
2 >>> data = [12,34,23,32,46,96,13]
>>> cl = HierarchicalClustering(data, lambda x,y: abs(x-y))
>>> cl.getlevel(10)
[[[96], [46], [12, 13, 23, 34, 32]]
>>> cl.getlevel(5)
7 [[96], [46], [12, 13], [23], [34, 32]]

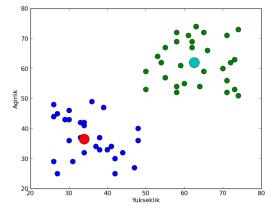
```

16.2 python-cluster ile KMeans Clustering

```

from cluster import KMeansClustering
from cluster import centroid
3 data = []
for i in range(30):
    x1=np.random.randint(25,50); x2=np.random.randint(25,50)
    y1=np.random.randint(60,85); y2=np.random.randint(60,85)
    data.append((x1, x2)); data.append((y1, y2))
8
cl = KMeansClustering(data); clusters = cl.getclusters(2)
x_val=[x[0] for x in clusters[0]];y_val=[x[1] for x in clusters[0]]
plot(x_val,y_val,'o')
x_val=[x[0] for x in clusters[1]];y_val=[x[1] for x in clusters[1]]
13 plot(x_val,y_val,'o')
center1=centroid(clusters[0]); center2=centroid(clusters[1])
plot(center1[0],center1[1],'o', markersize=20)
plot(center2[0],center2[1],'o', markersize=20)
xlabel('Yukseklik'); ylabel('Agirlik')

```



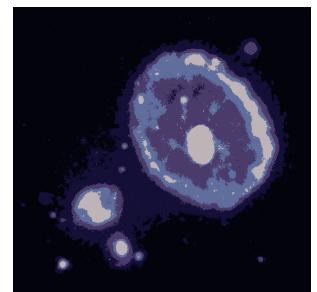
16.3 SciPy.cluster ile KMeans Clustering

```

import cv2
import numpy as np
3 from scipy.cluster import vq

img = cv2.imread('Cartwheel.jpg'); z = img.reshape((-1,3))
k = 6 ; center,dist = vq.kmeans(z,k)
code,distance = vq.vq(z,center); res = center[code]
8 res2 = res.reshape((img.shape)) ; cv2.imshow('res2',res2)
cv2.waitKey(0); cv2.destroyAllWindows()

```



16.4 PANDAS

16.4.1. CSV Dosyalar

```
1 from pandas import read_csv  
2 csv_data = read_csv('veriler.csv')  
3 csv_data = csv_data.values  
4 csv_data[:4]
```

16.4.2. EXCEL Dosyalar

```
1 from pandas import read_excel  
2 veri=read_excel('sonuc.xls','Sheet1')  
3 veri_data = veri.values  
4 veri_data[:4,:2]  
5 veri  
6 veri_data
```

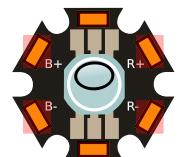
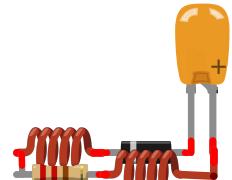
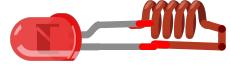
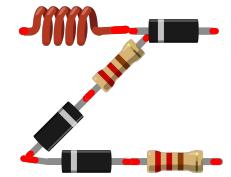
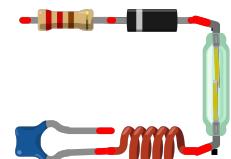
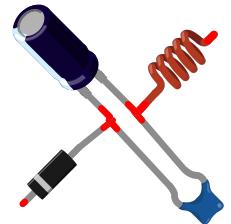
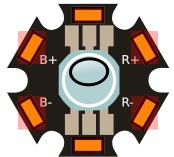
16.5 Ekler

```
a=<tr><td>34.4</td></tr>"  
a.split('<tr><td>')[1].split('</td></tr>')[0]  
import random; baz = list('ATGC'); random.choice(baz)
```

Akademik Bilişim Kursu III. Gün

LABORATUVARDA LINUX

Bu kursta laboratuvara ölçüme ve işlem kontrolü amaçlı kullandığımız cihazları Linux ile kullanmayı öğreneceğiz. Bilgisayar-Cihaz haberleşmeleri, elektronik dizayn uygulamaları, Mikrodenetleyiciler ve programlama ve grafiksel görüntüleme araçları konusunda temel düzeyde bilgi ve beceri kazandırmaya yönelikdir. Elektronik konusunda da temel bilgilere sahip olacağız.



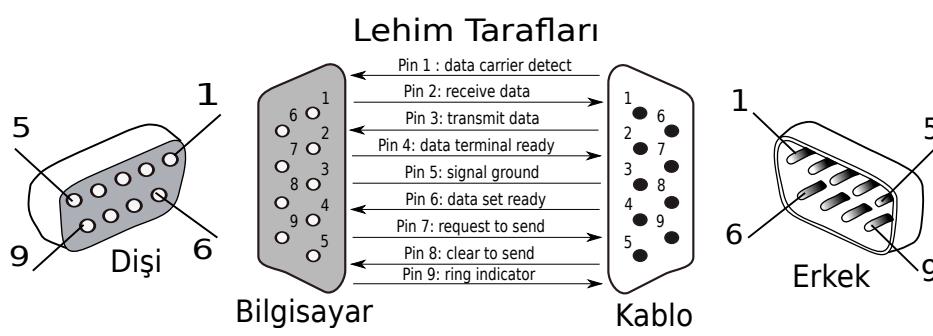
Created with  Fritzing.org

17.1 USB Bağlantıları

Pin	Sinyal	Renk	Tanımlama	Kablo	Cihaz
1	VCC	Kırmızı	+5V		
2	D-	Beyaz	Data-		
3	D+	Yeşil	Data +		
4	GND	Siyah	Toprak		

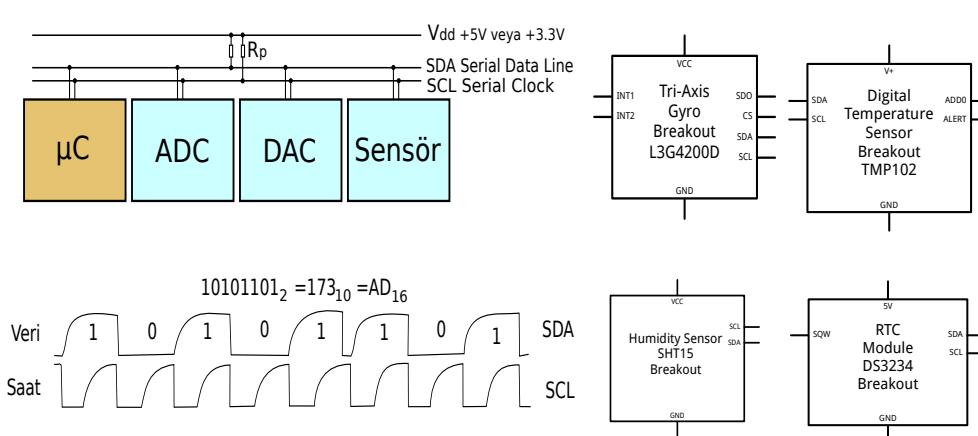
USB, (Universal Serial Bus) veya Evrensel Seri Veriyolu, dış donanımlar ile bilgisayar arasında seri haberleşme sağlayan bir bağlantı biçimidir.

17.2 RS232



RS232 Dış donanımlar ile bilgisayar arasında seri bağlantı amaçlı kullanılan bir iletişim şeklidir. Modern bilgisayarlarda artık RS232 portu bulunmamaktadır. Endüstriyel ve bilimsel cihazlarda hala RS232 portu üzerinden haberleşme yapmaktadır.

17.3 I²C



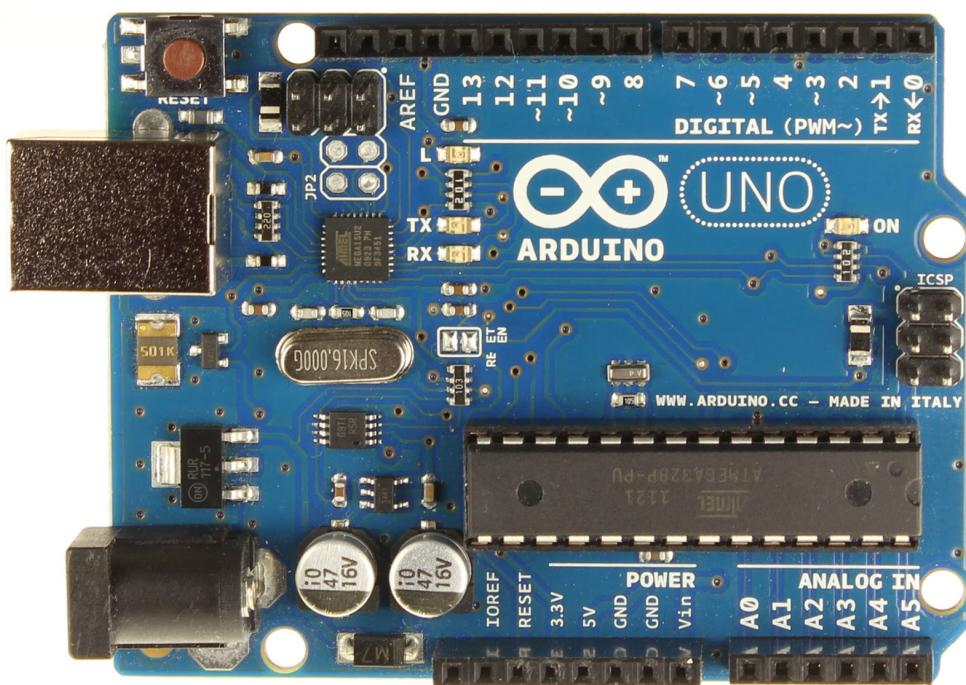
I²C Çevre birimlerini anakart, gömülü sistem veya mikro denetleyiciye bağlamak için kullanılan toprağa referanslı, çoklu iletişimine olanak sağlayan iki tel üzerinden haberleşme sağlayan seri veriyoludur. Sensörler, ADC veya DAC çipleri çoğunlukla mikrodenetleyiciye I²C ile bağlıdır.

17.4 Mikrodenetleyici

Bir mikrodenetleyici (MCU veya μ C), bellek ve merkezi işlem birimi dışında diğer birçok elektronik devre bileşenlerini içerisinde barındırır.

Mikrodenetleyiciler analog-dijital dijital-analog çeviriciler de barındırmaktadır. Küçük boyutlu, düşük güç tüketimi, düşük maliyet nedeniyle çevremizdeki hemen her elektronik cihazda kullanılmaktadır.

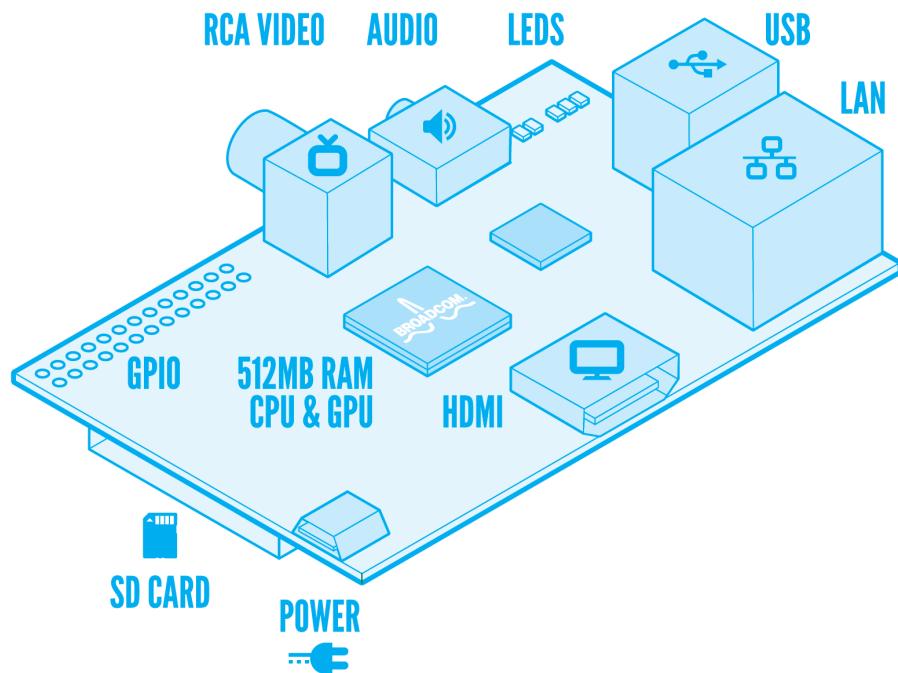
Kursda Arduino Uno programlama ve Linux ile haberleşme örnekleri verilecektir.

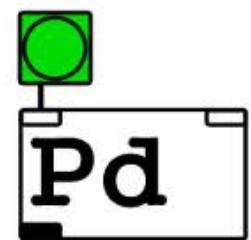


17.5 Gömülü Sistemler

Gömülü sistem, kişisel bilgisayarlardan farklı olarak, önceden tanımlanmış görevleri yerine getiren ve mikroişlemci ve mikrodenetleyici içeren, düşük maliyet ve küçük boyutlarda üretilen sistemlerdir. Kursta Raspberry Pi tanıtımı yapacaktır.

RASPBERRY PI MODEL B





Gerçek zamanlı görüntü ve ses sinyali işleme ve çevre birimleri arasında veri alışverişi için kullanılabilecek ve her platformda çalışan bir uygulamadır. Herşeyden önce açık kaynak ve özgür yazılım olması çok önemlidir. PureData'ya görsel programlama ortamı dersek yanlış olmaz. PureData Ubuntu paketleri içerisinde yer alırken PureData'nın genişletilmiş sürümünü kurmak için aşağıdaki şekilde komutlar girilmelidir.

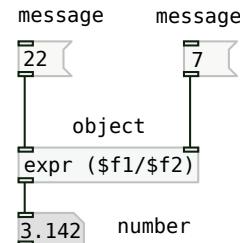
```
sudo add-apt-repository ppa:eighthave/pd-extended
sudo apt-get update
sudo apt-get install pd-extended
```

18.1 Pd-Extended ile Örnek Uygulama

\$ pd-extended

komutu ile puredata'yı çalıştırdıktan sonra File -> New ile yeni bir çalışma sayfası açınız. Put menüsünden Message, Number ve Object ekledikten sonra, yandaki şekildeki gibi objeleri birbirine bağlayınız. Nesnelerin yerlesimi önemli değildir, sadece görünüm açısından önemlidir. Nesneleri mouse tuşu ile tutarak yerlerini değiştirebilirsiniz. Ctrl+e ile Edit modundan çıkış sayıları değiştirdiğinizde iki sayının bölümü alttaki number nesnesi içerisinde belirecektir. Ctrl+e tuşlarına basarak "Edit" ile "Çalıştırma" modu arasında geçiş yapabilirsiniz.

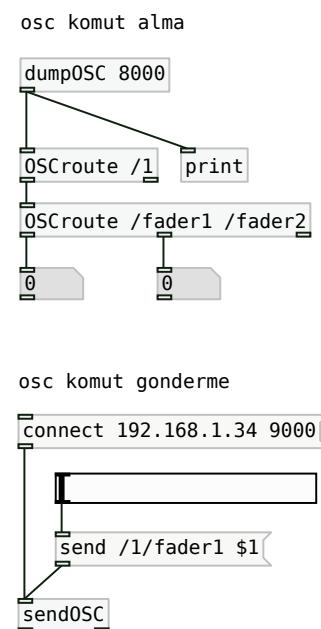
PureData ile ilgili bu kitapda sadece veri aktarımı ve sinyal işleme konularında örneklerde yer vereilecektir. PureData ile ilgili görsel ve metinsel kaynaklara internet üzerinden ulaşabilirsiniz.



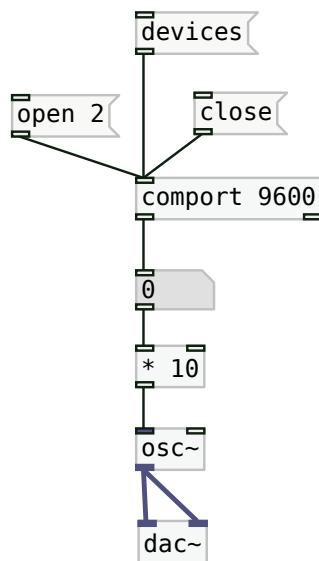
18.2 OSC ile Haberleşme

OSC (Open Sound Control) elektronik çalgılar arasında iletişim sağlamak amacıyla geliştirilmiş bir protokolmasına rağmen esnek bir yapıda olması nedeniyle laboratuvara kendi geliştirdiğimiz cihazlar arasında haberleşme yaptırmak amacıyla da kullanabiliriz. OSC mesajlarının iletimini sadece bilgisayar ağları üzerinden iletimine değineceğiz.

OSC ile veri gönderme ve alma örneğini akıllı telefon ile uygulayacağız. Android veya iPhone Marketten TouchOSC uygulamasını indiriniz. Ayarlar kısmında host sunucu olarak PC veya Labtopunuzun IP numarasını giriniz. Alma portu için i(incoming) 9000, gönderi (outgoing) portu için 8000 ön tanımlı olarak gelecektir. Farklı ise değiştirebilirsiniz. PC de ise veri 9000 inci porttan gönderilirken, veri 8000 inci porttan alınmaktadır. Cep telefonunuzda ekrandaki şekillerle oynadığınızda veriler PureData'nın konsol ekranında görünecektir. PC den veri gönderdiğinizde ise (burada /1/fader1 1) 1 numaralı sürgü en yukarıya çıkacaktır.



18.3 COM Port ile Seri Haberleşme



Herhangi bir seri cihazı (USB, RS232, Bluetooth) Linux kurulu bir bilgisayara taktığınızda, bu cihazın sistem tarafından tanınıp tanınmadığını anlamadan iki yöntemi vardır.

`$ dmesg`

komutu girilerek sistem takılan cihazın herhangi bir mesaj üretip üretmediğine bakabilirsiniz. Diğer bir komut da;

`$ lsusb`

komutudur. Bu komut da usb ye bağlı cihazların listesini verecektir.

Birer saniye aralıklarla 0 dan 100'e kadar sayı gönderen bir Arduino mikro denetleyiciden gelen sayıları okuyan bir pd-extended uygulaması aşağıda verilmiştir.

Yandaki şekildeki gibi pd-extended patch dosyası oluşturunuz. Devices mesaj kutusuna tıklandığında konsolda bilgisayarınıza bağlı usb veya rs232 com bağlantılarını listesini alacaksınız. Bu listeden bağladığınız seri cihazın numarasını edit modunda (Ctrl+e) open komutunun yanına yazınız. Ctrl+e ile edit modundan çıktıktan sonra open komutuna tıklayarak bağlantıyı gerçekleştiriniz. Bundan sonra seri cihazdan gelen sayılar frekansa çevrilerek hoparlörünüzden ses olarak duymanızı sağlayacak.

Arduino Dosyası

```

void setup() {
Serial.begin(9600);
}

void loop(){
for(int i=0; i<100;i++){
Serial.write(i);
delay(1000); }
}
  
```

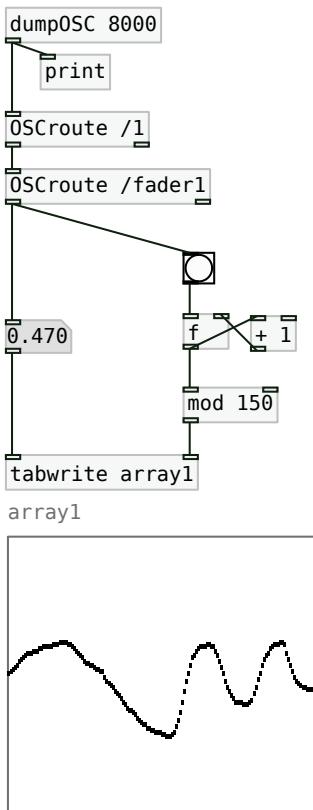
Pd-Extended Dosyası

```

|open 2<
|
|comport 9600|
|
|print |
  
```

18.4 Gerçek Zamanlı Grafik

Pd-Extended ile seri porttan veya ağdan alınan verileri gerçek zamanlı grafiğe geçirmek için tabwrite ve array nesnelerini kullanacağız. Yanda verilen şema cep telefonunda kurulu TouchOSC uygulamasında birinci penceredeki birinci kaydırma çubuğundan gelen verileri grafiğe geçirmektedir.



19.1 Seri Porttan Veri Okuma

```
1 >>> import serial; ser = serial.Serial("/dev/ttyACM0", 9600)
>>> while 1:
...     ser.readline()
'1 Hello world!\r\n'
'2 Hello world!\r\n'
```

19.2 Seri Porta Veri Gönderme

```
>>> import serial ; ser = serial.Serial("/dev/ttyACM0", 9600)
>>> ser.write('5'); ser.write("\xFF\x09\x03")
```

19.3 Python İle Grafiksel Arayüz Oluşturma

```
from Tkinter import * ;deger=50; top = Tk()
def geri():
    global deger ; print('Geri...') ; deger=deger-1
    sayac.set(deger)
def ileri():
    global deger ; print('İleri...') ; deger=deger+1
    sayac.set(deger)
8 B = Button(top, text ="Geri", command = geri)
B.pack({"side": "left"})
sayac = IntVar(); E = Entry(top, textvariable=sayac)
E.pack( {"side": "left"}); sayac.set(deger)
C = Button(top, text ="İleri", command = ileri)
C.pack({"side": "left"}); top.mainloop()
13
```

19.4 Azalan Coğalan Işık

```
int incomingByte = 0; int led = 9;
2 void setup() {
Serial.begin(9600);pinMode(led, OUTPUT); }
void loop() {
if (Serial.available() > 0) {
incomingByte = Serial.read(); analogWrite(led, incomingByte);
7 } }
```

19.5 Seri Donanım

```

1  /* RX digital pin 10 -> Cihaz TX  TX digital pin 11 -> Cihaz RX */
2
3 #include <SoftwareSerial.h>
4 SoftwareSerial mySerial(10, 11); // RX, TX
5 byte CMD_SET_AUTO_MODE[] =
6 {0xFF, 0x01, 0x09, 0x87, 0x01, 0x03, 0x02, 0x00, 0x10, 0x20, 0x30, 0x40, 0x37};
7 void setup()
8 { Serial.begin(9600); mySerial.begin(19200);
9   mySerial.write(CMD_SET_AUTO_MODE, sizeof(CMD_SET_AUTO_MODE)); }
10 void loop()
11 { if (mySerial.available()) Serial.println(mySerial.read(),HEX);
12   if (Serial.available()) mySerial.write(Serial.read()); }

```

19.6 AD7746 Capacitive Sensor

AD7746 Capacitive Sensor: Accuracy: 4 fF (yes femto) Linearity: 0.01% Common-mode (not changing) capacitance up to 17 pF Full-scale (changing) capacitance range: 4 pF Tolerant of parasitic capacitance to ground up to 60 pF

```

1 #include <Wire.h>
2 #define SLAVEWRT 0x48 // address of device during write cycle
3 #define SLAVERD 0x48 // address of device during read cycle
4 int mySensor=0;//GND->GND, 3.3V->3.3V, SDA->AnlgIn4, SCL->AnlgIn 5
5 void setup()
6 {
7   Wire.begin(); Serial.begin(9600);
8   Wire.beginTransmission(SLAVEWRT); Wire.write(0xBF);
9   Wire.endTransmission(); Wire.beginTransmission(SLAVEWRT);
10  Wire.write(0x0A); Wire.write(0x01 | (7 << 3));
11  Wire.endTransmission(); Wire.beginTransmission(SLAVEWRT);
12  Wire.write(0x07); Wire.write(0x80);
13  Wire.endTransmission(); Wire.beginTransmission(SLAVEWRT);
14  Wire.write(0x09); Wire.write(0x08);
15  Wire.endTransmission(); Wire.beginTransmission(SLAVEWRT);
16  Wire.write(0x0B); Wire.write(0x00 | 39);
17  Wire.endTransmission(); Wire.beginTransmission(SLAVEWRT);
18  Wire.write(0x0A); Wire.write(0x01 | (7 << 3));
19  Wire.endTransmission();
20 }
21 void loop() // main program begins
22 {
23   Wire.beginTransmission(SLAVERD); Wire.write(0x00);
24   Wire.endTransmission(); Wire.requestFrom(SLAVERD,4);
25   byte data[2]; int i = 0;
26   while(Wire.available()){
27     char c = Wire.read(); data[i] = c;i++; }
28   mySensor = -((data[0]<<16)+(data[1]<<8)+data[2]) / 12;
29   Serial.println(mySensor); delay(50);
30 }

```

İÇİNDEKİLER

i	GNU LINUX	1
1	LINUX	3
2	UNIX	5
3	BASH KABUĞU	9
4	METİN EDITÖRLERİ	11
5	MAKALE VE TEZ	15
6	GRAFIK	17
ii	BİLİMSEL PROGRAMLAMA	19
7	PYTHON	21
8	NUMPY	23
9	PYTHON İLE GRAFIK	25
10	SEMBOLİK MATEMATİK	27
11	OPTIMİZASYON	29
12	İSTATİSTİK	31
13	FAKTORİYEL TASARIM	33
14	PCA	35
15	GÖRÜNTÜ İŞLEME	37
16	VERİ MADENCİLİĞİ	39
iii	LABORATUVARDA LINUX	41
17	ELEKTRONİK	43
18	PUREDATA	45
19	LINUX LAB	47