

PRAKTIKUM IZ SOFTVERSKIH ALATA U ELEKTRONICI

2018/2019

Predrag Pejović

12.01.2019, 13:26

Linkovi na primere:

OS: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-3-2018.zip>

L^AT_EX 1: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-4-2018.zip>

L^AT_EX 2: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-5-2018.zip>

L^AT_EX 3: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-6-2018.zip>

GNU Octave: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-8-2018.zip>

gnuplot: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-10-2018.zip>

Maxima: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-11-2018.zip>

Python 1: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-12-2018.py>

Python 2: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-13-2018.zip>

PyLab: <http://tnt.etf.bg.ac.rs/~oe4sae/primeri-14-2018.zip>

SymPy: <http://tnt.etf.bg.ac.rs/~oe4sae/prim-sympy.py>

SymPy-DEq: <http://tnt.etf.bg.ac.rs/~oe4sae/prim-sympy-dj.py>

Lica (i ostali podaci o predmetu):

PRAKTIKUM IZ SOFTVERSKIH ALATA U ELEKTRONICI 2018

- ▶ Predrag Pejović, peja@etf.rs, 102 levo, <http://tnt.etf.rs/~peja>
- ▶ Strahinja Janković
- ▶ sajt: <http://tnt.etf.rs/~oe4sae>
- ▶ cilj: savladavanje niza programa koji se koriste za svakodnevne poslove u elektronici (i ne samo elektronici ...)
- ▶ svi programi koji će biti obrađivani su slobodan softver (free software), legalno možete da ih koristite (i ne samo to) gde hoćete, kako hoćete, za šta hoćete, koliko hoćete, na kom računaru hoćete ...
- ▶ literatura ... sve sa www, legalno, besplatno!
- ▶ zašto svake godine (pomalo) updated slajdovi?

Prezentacije predmeta

- ▶ engleski
- ▶ srpski, kraća verzija
- ▶ engleski, prezentacija i animacije
- ▶ srpski, prezentacija i animacije

Evolucija i društvene nauke

- ▶ teorijski, počinje 1940s
- ▶ praktično, kod nas možda 1970s
- ▶ 1980s koristimo računar
- ▶ 1990s koristimo računar u svakodnevnom radu
- ▶ 2000s sve radimo pomoću računara
- ▶ 2010s potpuno zavisimo od računara
- ▶ PC, mobile, cloud, phone, tablet, ...
- ▶ nema više PC? za koga?
- ▶ ubrzava se ...

Pravni i ekonomski problemi

- ▶ Pravni problemi
 - ▶ licenciranje (program koji ste kupili je vaš?)
 - ▶ **MORAMO biti 100% legalni!!!**
- ▶ Ekonomski problemi
 - ▶ troškovi (i ne samo to! ↓)
 - ▶ navike i zavisnost
 - ▶ problemi sa proprietary formatima (primer: bas bih voleo opet da procitam svoj doktorat u elektronskoj formi)
 - ▶ bezbednost podataka!
 - ▶ način naplate ...
 - ▶ opravdanost naplate ...
 - ▶ free (slobodni) software?
 - ▶ besplatno (ako tako hoćete) i jos mnogo više!

A šta se tačno radi u predmetu, koji programi?

1. uvod (upravo slušate): organizacija nastave + (FS: tehnička, ekonomska i pravna pitanja, kako to uopšte postoji?) ($\approx 1 w$)
2. operativni sistem (GNU/Linux, Ubuntu), komandna linija (!), shell scripts, ... ($\approx 1 w$)
3. nastavak OS, snalaženje, neki IDE kao ilustracija i vežba, jedan Python i jedan C program ... ($\approx 1 w$)
4. \LaTeX i $\LaTeX 2_{\epsilon}$ ($\approx 3 w$)
5. Xcircuit ($\approx 1 w$)
6. probni kolokvijum ... (= 1 w)
7. prvi kolokvijum ...
8. GNU Octave ($\approx 1 w$)
9. gnuplot ($\approx (1 + \epsilon) w$)
10. wxMaxima ($\approx 1 w$)
11. drugi kolokvijum ...
12. Python, IPython, PyLab, SymPy ($\approx 3 w$)
13. treći kolokvijum ...

Problemi u radu sa računarima ...

- ▶ svaki program zahteva obuku ...
- ▶ čitati uputstvo? dostupna literatura? rano odustajanje ...
- ▶ najbrže i najbolje se uči ako neko pokaže ("hands-on approach")
- ▶ pomoćnik-savetodavac dostupan?
- ▶ Google pomoćnik? Youtube pomoćnik?
- ▶ koji program odabrati? stvaranje zavisnosti (navike + old files)? ozbiljna odluka!
- ▶ da li računar pomaže ili odmaže? kada se vraća naučeno?
- ▶ strahovito brze promene, novi programi ...
- ▶ prilagođavanje: psihologija, ergonomija, namere autora programa (vagon primer) ...
- ▶ rezultati: tehnofobija illi zavisnost
- ▶ cilj: **računar treba da služi nama, ne mi njemu**

Kako je nekada bilo ... ("učiteljica života")

- ▶ mainframe računari
- ▶ software i hardware se prodavali zajedno
- ▶ računari bili ekskluzivan proizvod, skup
- ▶ mnogo manje korisnika
- ▶ mnogo manje komunikacije između korisnika
- ▶ okruženje bilo uglavnom naučno ili visoko poslovno
- ▶ poslovne primene uz razvoj sopstvenih programa, custom design
- ▶ dominantna zaštita poslovna tajna
- ▶ ogromne mogućnosti, pps, strahovito brz razvoj
- ▶ društvene implikacije, nestanak i nastanak velikog broja poslova, promena sistema vrednosti, pojava mladih ljudi na uticajnim pozicijama ...

Kako je danas? (posle ≈ 1985)

- ▶ PC revolucija (prošla; prošla?)
- ▶ jeftino, dostupno, (modularno ??)
- ▶ sklopivo, prilagodljivo korisniku (?)
- ▶ izaberete komponente (?), sklopite računar (?), šta dalje?
- ▶ softver postao zasebna komponenta računara, posebno se kupuje (upravo ovde priča počinje ...)
- ▶ operativni sistem, početak
- ▶ korisnički softver, zavisno od primene računara
- ▶ softver je intelektualni, nematerijalni proizvod ...
- ▶ nematerijalni proizvodi su **veoma** specifični, upotrebom se ne troše!
- ▶ rezultat: monopoli i „verski ratovi“
- ▶ ideologizacija i manipulacija ...
- ▶ http://en.wikipedia.org/wiki/Samizdat:_And_Other_Issues_Regarding_the_%27Source%27_of_Open_Source_Code

Specifičnosti „intelektualnih“ (nematerijalnih) proizvoda 2

(navedene osobine se uglavnom odnose na softver, manje na ostale intelektualne proizvode)

- ▶ ogroman komercijalni potencijal
- ▶ potencijal za stvaranje zavisnosti (izuzetno opasno!)
- ▶ potencijal za uslovljavanje
- ▶ problemi vezani za privatnost
- ▶ primeri:
 - ▶ Phil Zimmerman, Boulder, CO, 1991, PGP
 - ▶ Mark Shuttleworth (1973), Thawte, digital certificates and Internet security
 - ▶ Mark Shuttleworth, Canonical, 2004, biće još reći ...
- ▶ sjajna osnova za izgradnju „social machinery“ ni oko čega ...
- ▶ u čemu ima mnogo interesa ... koji se brane!

Copyright

- ▶ štiti konkretan proizvod, ne ideju
- ▶ dugo traje
- ▶ produžava se
- ▶ širi se domen
- ▶ sve copyrightable automatski je copyrighted (negde ...)
- ▶ copyright notice, „all rights reserved“ (ne košta!)
- ▶ dominantna zaštita za software u nizu pravnih sistema ...
- ▶ pošto se software kopiranjem umnožava, „proizvodi“
- ▶ ne sprečava da neko napiše nov program iste funkcionalnosti!

Trade Secret (poslovna tajna)

- ▶ ne košta
- ▶ traje dok traje tajna
- ▶ nikog ne sprečava da otkrije isto
- ▶ zaštita kroz NDA
- ▶ negde se može primeniti, negde ne
 - ▶ primer: program u interpretativnom jeziku

Specifičnosti „intelektualnih“ (nematerijalnih) proizvoda 1

- ▶ značajni troškovi razvoja
- ▶ proizvod se upotrebom ne troši, traje
- ▶ potražnja se spontano ne obnavlja
- ▶ za zaradu neophodan novi razvoj ili vremenski ograničena licenca
- ▶ misaoni eksperiment: softver koji se kviri
- ▶ prestaje „podrška“
- ▶ troškovi umnožavanja zanemarljivi
- ▶ troškovi distribucije zanemarljivi
- ▶ kako organizovati proizvodnju da bi se ostvario profit?
- ▶ karakteristike veoma slične stvaralaštvu u nauci, oblast slična matematici ...
- ▶ postoje programeri koji vole da programiraju ...
- ▶ ... dok ne dodje GUI!

„Intelektualna svojina“ i njena „zaštita“

pokušaj da ideje dobiju tretman materijalnih objekata, a time i svog vlasnika

forme „intelektualne svojine“:

1. **copyright**
2. **patent**
3. **trade secret**

bitno se razlikuju!!!

Patent

- ▶ štiti ideju
- ▶ kratko traje (10-20 godina)
- ▶ kratko? za softver kratko?
- ▶ širi se domen (predmet, metod, nova primena, organizam, geni ...)
- ▶ zahteva prijavu i odobrenje patenta, košta!
- ▶ David Pressman, „Patent it Yourself,“ Nolo Press
- ▶ offensive rights
- ▶ samo 10% patenata isplati troškove (???)
- ▶ kritike, 19. vek, „Patent Absurdity“
<http://patentabsurdity.com/>
- ▶ nečitki patenti, promenjen smisao
- ▶ pojam „očigledno“
- ▶ kako to funkcioniše u praksi ...

Softver, terminološke zabune

- ▶ Free Software
- ▶ Open Source Software
- ▶ Freeware
- ▶ Shareware

Free Software

- ▶ Richard Mathew Stallman, 1983.
- ▶ free as in free speech, not free beer (slobodan / besplatan)
- ▶ uvek je neko „platio“ razvoj programa
- ▶ korektan prevod „slobodan softver“
- ▶ definicija (po RMS):
 - ▶ Freedom 0: da se izvršava program, bez ograničenja
 - ▶ Freedom 1: da se proučava i menja program: potreban source code
 - ▶ Freedom 2: da se distribuiraju kopije programa
 - ▶ Freedom 3: da se distribuiraju modifikovane verzije programa
- ▶ kad licenca ovo dopušta software je free
- ▶ FSF, <http://www.fsf.org>
- ▶ FSFE, <http://fsfe.org/>
- ▶ osim praktičnih, pokreću i etička pitanja

Podela softvera

- ▶ free software (slobodni softver)
 - ▶ licenca dopušta navedene četiri slobode
 - ▶ izvorni kod (source code) uvek dostupan
 - ▶ ima više licenci koje to omogućuju
 - ▶ primeri licenci: GNU GPL (GNU/Linux) i BSD (BSD, Python, Spice)
- ▶ proprietary software (vlasnički softver)
 - ▶ licenca ne dopušta bar jednu od navedene četiri slobode
 - ▶ izvorni kod može biti dostupan
 - ▶ izvorni kod obično nije dostupan, poslovna tajna
 - ▶ korisnik baš i ne zna šta program radi na njegovom računaru
 - ▶ maksimiranje profita utiče na software design, primeri ...

Freeware

- ▶ proprietary, ali besplatan softver
- ▶ besplatan iz nekog razloga
 - ▶ reklame
 - ▶ prikupljanje informacija
 - ▶ stvaranje navike kod korisnika
 - ▶ testiranje programa (besplatno)
 - ▶ testiranje tržišta
 - ▶ želja za sakrivanjem dela koda ili podataka
 - ▶ ...
- ▶ obično nepoznat source code
- ▶ proučiti motive pre upotrebe, razmisliti
- ▶ primer: [LTSpice](#)

GNU

- ▶ 1983. ili 1984., GNU project, RMS, MIT
- ▶ recursive acronym, GNU's Not Unix
- ▶ da bude i ostane free
- ▶ nije baš bilo bez pomoći sa strane, prećutna podrška MIT ...
- ▶ kako da ostane free?
- ▶ izmišljen [copyleft](#) (jako važno! biće još reči ...)
- ▶ <http://copyleft.org/>
- ▶ zašto baš Unix?

novi i neobično, ...

- ▶ neprijatelji? brojni!
 1. interesni, racionalni
 2. ideološki, iracionalni
- ▶ korisnici?
- ▶ motivi contributors?
- ▶ treba li programeri da budu plaćeni?
- ▶ poslovni modeli?
- ▶ „pustite nas da radimo!“
- ▶ “social machinery”
- ▶ konkurencija (lojalna, neloyalna?)
- ▶ prošlost kod nas (do ≈ 2003.): sve je bilo freeware, jako malo free software

Open Source Software

- ▶ Bruce Perens i Eric Steven Raymond (ESR)
- ▶ “The Cathedral and the Bazaar”
- ▶ Open Source Initiative, www.opensource.org
- ▶ u osnovi, to je free software
- ▶ ideja: popraviti marketing (ne zamerati se)
- ▶ podržao [Linus Torvalds](#)
- ▶ u početku uspeh
- ▶ softver kome se zna izvorni kod MOŽE biti proprietary software!!!
Primer: [FFTW](#)
- ▶ gotovo uvek je (open source = free) software
- ▶ FOSS (Free and/or Open Source Software)
- ▶ ozbiljan sukob sa FSF, koristili i dalje GPL

Shareware

- ▶ problem kod reči “sharing”
- ▶ softver koji se besplatno distribuira
- ▶ manje slobodan nego freeware
- ▶ obično služi za promociju programa
- ▶ ograničena funkcionalnost i/ili vremensko ograničenje

Unix, UNIX (i C)

- ▶ Unix ...
- ▶ 1969, AT&T, Ken Thompson, Dennis Ritchie, Brian Kernighan, Douglas Mclroy, and Joe Ossanna
- ▶ portability
- ▶ modularity
- ▶ Unix philosophy
- ▶ vrlo povoljna osnova za GNU!
- ▶ C kompajler i modul po modul, ...

Dennis Ritchie, #1

Following Ritchie's death, computer historian Paul E. Ceruzzi stated:

"Ritchie was under the radar. His name was not a household name at all, but ... if you had a microscope and could look in a computer, you'd see his work everywhere inside."

Linux

- ▶ do 1991. skoro sve gotovo
- ▶ nedostajao kernel OS
- ▶ RMS očekivao Hurd
- ▶ Hurd ne baš sjajno napredovao
- ▶ Linus Torvalds, University of Helsinki → Linux
- ▶ konačno sve gotovo, free OS
- ▶ počele distribucije, ...
- ▶ „nesporazumi oko imena“
- ▶ danas uglavnom GNU/Linux ili GNU+Linux

Licence ...

- ▶ [Understanding Open Source and Free Software Licensing](#)
- ▶ [GPL, General Public License, FSF, copyleft, ...](#)
- ▶ [BSD License](#), permissive free software license
- ▶ [MIT License](#)
- ▶ dvojno licenciranje moguće, [FFTW](#)
- ▶ [CeCILL](#)
- ▶ <http://www.gnu.org/licenses/license-list.html>

MIT license

http://en.wikipedia.org/wiki/MIT_License

Copyright (C) <year> <copyright holders>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Dennis Ritchie, #2

Views on computing

In an interview from 1999, Dennis Ritchie clarifies that he sees Linux and BSD operating systems as a continuation of the basis of the Unix operating system, and as derivatives of Unix:

"I think the Linux phenomenon is quite delightful, because it draws so strongly on the basis that Unix provided. Linux seems to be among the healthiest of the direct Unix derivatives, though there are also the various BSD systems as well as the more official offerings from the workstation and mainframe manufacturers."

POSIX

Ličnosti

- ▶ [Richard Mathew Stallman \(RMS\)](#)
- ▶ [Eben Moglen](#)
- ▶ [Linus Torvalds](#)
- ▶ [\(Larry Lessig, Creative Commons\)](#)
- ▶ [Mark Shuttleworth](#)

Spice, BSD licenca, original

http://en.wikipedia.org/wiki/BSD_licenses

7/17/2007

Spice is covered now covered by the BSD Copyright:

Copyright (c) 1985-1991 The Regents of the University of California. All rights reserved.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.


IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Važno: Copyleft

- ▶ **Copyright:** "all rights reserved"
- ▶ **Copyleft:** "all rights reversed" ili "all wrongs reversed" ili "some wrongs reversed"
- ▶ cilj da se free software zaštiti, da ostane slobodan
- ▶ restriktivne licence
- ▶ „virusna licenca“
- ▶ [The GNU General Public License](#)
- ▶ [The GNU Lesser General Public License](#)
- ▶ [The GNU Affero General Public License](#)
- ▶ [The GNU Free Documentation License](#)

A šta je sa hardverom?

- ▶ „otvoreni hardver“
- ▶ **Arduino**: [Creative Commons Attribution Share-Alike License](#)

- ▶ ... kao i niz GPL licenci ...
- ▶ **Interaction Design Institute Ivrea**, 2001-2005
- ▶ ozbiljna igra: **RISC-V**, UC Berkeley, Krste Asanović
- ▶ BSD licenca
- ▶ **SiFive**
- ▶ ... upravo se događa, biće o ovome priče 13.10.2018, Vlada Milovanović, radio sa Borom Nikolićem ...
- ▶ **reakcija** ARM-a

Organizacija predmeta

- ▶ sasvim praktičan predmet, lak; lak (?)
- ▶ savladati ((samostalno) korišćenje) niza programa
- ▶ organizacija nastave: radionica, "hands-on approach"
- ▶ **Q: koliko ljudi za računarom?**
- ▶ **A: preferably 1!**
- ▶ maksimum 20 studenata, najbolje 10
- ▶ ocenjivanje: mali kolokvijumi, 3 kolokvijuma
- ▶ za koga predmet **nije**

Šta bi još moglo da se uvrsti u predmet?

- ▶ VirtualBox
- ▶ LibreOffice
- ▶ Scilab, Xcos, (Freemat, Euler, ...)
- ▶ Inkscape
- ▶ GIMP, ImageMagick
- ▶ LibreCAD
- ▶ Asymptote
- ▶ git
- ▶ Qucs
- ▶ ngspice
- ▶ KiCad
- ▶ R
- ▶ Julia
- ▶ SageMath ...

Linkovi ...

sajt predmeta:

<http://tnt.etf.bg.ac.rs/~oe4sae/>

edX Linux kurs (bez programa koje obrađuje PSAE):

<https://www.edx.org/course/linuxfoundationx/linuxfoundationx-lfs101x-introduction-1621>

Cloud?

- ▶ There is no cloud: it's just someone else's computer!
- ▶ **Who does that server really serve?**
- ▶ baš dobar koncept: ni softver ni vaši podaci nisu kod vas
- ▶ stvara se zavisnost
- ▶ nekome se isplati: isplati li se vama?
- ▶ **Service as a Software Substitute (SaaS)**
- ▶ svakako aktuelna tema ...

Program predmeta, teme

(ono što je realno moglo da se stigne prethodnih godina)

- ▶ Ubuntu, osnove i komandna linija, shell scripts
- ▶ C, Python, IDEs, kao ilustracija prethodnog
- ▶ \LaTeX
- ▶ Xcircuit
- ▶ GNU Octave
- ▶ gnuplot
- ▶ Maxima, wxMaxima
- ▶ **Python**, PyLab (NumPy, SciPy, Matplotlib, IPython), SymPy

Ispit, ocenjivanje

- ▶ kolokvijumi/ispiti sa literaturom
 1. dokumentacija 1/3 (1. kolokvijum)
 2. matematika 1/3 (2. kolokvijum)
 3. programiranje 1/3 (ispit)
- ▶ ne traži se memorisanje, već snalaženje
- ▶ na kolokvijumu/ispitu jedino nije dozvoljena saradnja među studentima
- ▶ Ocenjivanje po kolokvijumima:
 1. LaTeX (80) + Xcircuit (20)
 2. Beamer (20) + Octave (20) + gnuplot (30) + wxMaxima (30)
 3. Python (50) + PyLab (50) + SymPy (30, bonus)
- ▶ otvoreno za primedbe i sugestije (do određene granice) ...

HONOR CODE
CERTIFICATE



Jerry Cooperstein, Ph. D.
Training Program Director
The Linux Foundation

Clyde Seepersad
General Manager, Training
The Linux Foundation

Predrag Pejovic

successfully completed and received a passing grade in

LFS101x: Introduction to Linux

a course of study offered by LinuxFoundationX, an online learning initiative of The Linux Foundation through edX.

HONOR CODE CERTIFICATE
Issued August 15th, 2014

Verify the authenticity of this certificate at:
<https://verify.edx.org/ver/0b294574-99408e8992ad07b1ffcc5>

Linkovi za zainteresovane ...

[Richard Matthew Stallman, TEDx, Geneva, 2014.](#)

[Eben Moglen, kako je sve počelo ...](#)

[Eben Moglen, gde smo sada, 2013.](#)

[Revolution OS, dokumentarni film iz 2001.](#)

[O patentima ...](#)

operativni sistem

— part 1 —

operativni sistem

- ▶ služi da bi pokretali i koristili programe
- ▶ win, Mac OS X, GNU/Linux, BSD, Solaris, OpenIndiana, ...
- ▶ GNU/Linux, distribucije:
 - ▶ **Ubuntu**
 - ▶ **Linux Mint**
 - ▶ **fedora**
 - ▶ **debian**
 - ▶ **openSUSE**
 - ▶ **KNOPPIX** (sjajno za probu i popravke, live DVD, live USB disk, "live medium")
 - ▶ ... **distrowatch**
 - ▶ i jedna knjiga: **linux from scratch**
- ▶ **Ubuntu** ...
- ▶ ... i bas za hardline **free-distros**

Ubuntu

- ▶ www.ubuntu.com
- ▶ desktop, server, Kubuntu, Xubuntu, Lubuntu, Edubuntu, Ubuntu GNOME, Ubuntu MATE ... menja se ... (Gnubuntu, Ubuntu-libre, Gobuntu)
- ▶ dnload, narezati DVD ili napraviti USB startup disk
- ▶ narezati? InfraRecorder, <http://infirarecorder.org/>, GPLv3
- ▶ pre instalacije **PROBATI**, live DVD, live USB disk, "live medium"
- ▶ može da se napravi USB startup disk, uputstvo ...
- ▶ Ubuntu, instalacije
 - ▶ samo Ubuntu
 - ▶ dual boot (dva diska? pazite!)
 - ▶ wubi, inside windows, prošlost?
 - ▶ **VirtualBox**, iso file, + some proprietary drivers ...

desktop environments

- ▶ kod windows samo win
- ▶ duga **istorija** pre toga, Xerox, Apple, ...
- ▶ kod GNU/Linux mnoštvo desktop environment mogućnosti
- ▶ popularni **GNOME**, **KDE**, **Xfce**, **LXDE**, **MATE**, **Cinnamon**, do skoro **Unity** ...
- ▶ svaki od desktop environments ima svoje motive, istoriju, izgled, ... **HIG**
- ▶ na kraju sve skoro isto, ali treba vremena da se to shvati ...
- ▶ možete da instalirate i štašta drugo: **GNOME**, **MATE**, **Cinnamon**, **Cairo**, ...
- ▶ sjajna zabava kada imate višak vremena

setting up the system ...

- ▶ instalira se operativni sistem ...
 - ▶ ... proprietary drivers (uglavnom graphics i wlan) ...
 - ▶ ... proprietary codecs ...ogg? <http://xiph.org/>
 - ▶ ubuntu-restricted-extras
- ▶ programi se instaliraju iz repository ...
- ▶ za instalaciju potreban **jedan** disk ili USB flash drive i **Internet**
- ▶ za probu sjajno ako imate stari racunar/laptop
- ▶ moje iskustvo sa instalacijama ...
- ▶ a sada pretpostavimo da je sistem instaliran ...

desktop environment ... DEFAULT!

- ▶ zašto default?
- ▶ **insistiramo na defaults**
- ▶ customize sami ... na **SVOM** nalogu i/ili računaru
- ▶ **pitanje pristojnosti** ...
- ▶ do sada nije bilo problema ...
- ▶ svoje podatke ponesite na USB flash drive ... ako su vam važni
- ▶ ne ostavljajte podatke na računaru, počistite ...
- ▶ rekoh, **pitanje pristojnosti** ...

desktop environment ...

- ▶ sistem boot-ovan
- ▶ let's log in ...
- ▶ launcher (application switcher) (jedina customization koju sam radio u odnosu na default, amazon out!)
- ▶ top menu bar
- ▶ indicators ...
- ▶ activities ...
- ▶ applications ...
- ▶ Desktop

launcher ...

- ▶ levi bar
- ▶ pokreće (lansira) aplikacije
- ▶ ostale aplikacije se dodaju (Add to Favorites) ili uklanjaju (Remove from Favorites), desno dugme miša
- ▶ ostalim aplikacijama se lako menja redosled
- ▶ aplikacija se pokrece sa single click
- ▶ pokrenuta aplikacija: tačka (puna)
- ▶ vežba: Octave, $2 + 2$
- ▶ vežba: wxMaxima $\int x^2 dx$

Show Applications

- ▶ pretražuje računar, i applications i files
- ▶ pokreće se sa single click dole levo
- ▶ Frequent / All
- ▶ name search

top menu bar ...

- ▶ levo, **Activities**
- ▶ sredina, time/date, notifications
- ▶ desno, system status and parameters
- ▶ sound
- ▶ Internet connection
- ▶ keyboard switcher
- ▶ settings
- ▶ lock, probati lock
- ▶ ...

razlike ...

- ▶ različiti desktop environments imaju različite nazive i različit raspored komandi
- ▶ primer: Linux Mint na Električnim merenjima
- ▶ ipak, ista funkcionalnost se ostvaruje
- ▶ potrebno kratko navikavanje i povremena potraga da se nešto pronađe
- ▶ pitanje ukusa o kojima nema smisla raspravljati ...

get acquainted, startovanje programa, ...

- ▶ updated?
- ▶ Octave, plot x^2 za $0 \leq x \leq 1$, ... kao Matlab, isto!
- ▶ **octave**, poznato?
- ▶ **octave-cli**, za old-fashioned
- ▶ wxMaxima, `solve(x^2=4, x)`, `wxplot2d(x^2, [x,-1,1])`
- ▶ gedit, iz Launcher-a
- ▶ terminal, **gnuplot**, `plot sin(x)`
- ▶ terminal, **ipython**, `a=2, b=3, a^b, a**b`
- ▶ terminal, **gedit**

vežba ...

- ▶ start Software Updater
- ▶ update
- ▶ start Ubuntu Software Center
- ▶ install GeoGebra
- ▶ run GeoGebra
- ▶ ne radi? uninstall!
- ▶ **sajt**, download, classic 6, install, radi!
- ▶ nacrtajte elipsu i još ponešto
- ▶ Add to Favorites
- ▶ Remove from Favorites
- ▶ pokrenete wxMaxima
- ▶ primer na wxMaxima ...
- ▶ Alt-Tab menja aplikacije

system settings ...

- ▶ System Settings
- ▶ pogledati Details
- ▶ u 18 **ne menjati** System Settings!
- ▶ rekoh, pitanje pristojnosti!

Files ...

- ▶ obično prva ikona, mada ne mora
- ▶ ime sve kaže
- ▶ dva prozora (taba), ctrl-t, može i više
- ▶ gase se klikom (desno je x, levo je ime)
- ▶ full path ctrl-l, izlaz sa esc
- ▶ ovo je otprilike sve za početak ...

file system ...

- ▶ Home Folder (Home Directory), ima svaki korisnik
- ▶ Ctrl/1, full path, Location, wheel-button
- ▶ subfolders, create your folder
- ▶ MSDOS: **folder**; UNIX, GNU/Linux: **directory** (mnogo ovakvih situacija)
- ▶ MSDOS: directory separator \; UNIX, GNU/Linux: directory separator /
- ▶ MSDOS: command option separator /; UNIX, GNU/Linux: command option separator -, --
- ▶ POSIX, UNIX, Linux, BSD, ... jedan tree, vратиću se ...
- ▶ win: svaka storage device ima tree, razlika!
- ▶ **moć navike!** (dobre i lose navike)
- ▶ komandna linija? već ste je sreli: terminal

komandna linija ...

- ▶ Terminal (izlaz: exit ili Ctrl/d)
- ▶ Alt/Ctrl F1 do F6, tty1 do tty6, neki je vaš prozor sada je to F2
- ▶ eto nas na komandnoj liniji
- ▶ jako dobra knjiga:
<http://sourceforge.net/projects/linuxcommand/>
- ▶ ali 544 strane!
- ▶ lako se čita!!! (tablet, prevoz, ...)

vrlo osnovne komande, ls

- ▶ uvek prva komanda, ls, ekvivalent za dir iz MSDOS-a
- ▶ `ls`, `ls -l`, `ls -a`, `ls -al`, `ls -lh`
- ▶ boje govore o tipu fajla, zelena je .exe ili .com ili .BAT
- ▶ `man ls`, `man -t ls`, `man -t ls >ls.ps`
- ▶ `ls`
- ▶ `ps2pdf ls.ps`
- ▶ `ls`
- ▶ `file ls.ps`
- ▶ `file ls.pdf`
- ▶ `evince ls.pdf`
- ▶ `evince ls.ps`
- ▶ postoji i evince za win,
<http://projects.gnome.org/evince/>
- ▶ `man convert` ...

cat, more, less ... biće još jednom

- ▶ `cat ls.ps`
- ▶ `cat ls.ps ls.ps`
- ▶ `man cat`
- ▶ ko je napisao cat ?
- ▶ `more ls.ps`
- ▶ `less ls.ps`
- ▶ less is more!

help na komandnoj liniji ...

- ▶ izlaz je q, ne esc
- ▶ `man ls` (manual)
- ▶ `ls --help` (help)
- ▶ `info ls` (GNU info system)
- ▶ `apropos`, `man apropos`
- ▶ `whatis`, `whatis apropos` (vrlo kratak opis)
- ▶ `whereis`, `whereis apropos` (vrlo kratak opis)

neke komande ...

- ▶ `date`
- ▶ `cal`
- ▶ `cal -help`
- ▶ `ncal`
- ▶ `df`
- ▶ `clear`
- ▶ `free`
- ▶ od sada: terminal na medium, File Manager, home, medium
- ▶ ideja: da vidite šta komandna linija radi, kako fajlovi i direktorijumi nastaju i nestaju

pokušaj prevare, cp, file

- ▶ `ls ls.ps`
- ▶ `cp ls.ps lspdf.pdf`
- ▶ `ls lspdf.pdf`
- ▶ `file lspdf.pdf`
- ▶ `cp ls.pdf lsps.ps`
- ▶ `ls lsps.ps`
- ▶ `file lsps.ps`

cat, gedit i redirekcija

- ▶ `gedit proba`
- ▶ sadržaj fajla po želji, npr dva reda teksta, save!!!
- ▶ `cat proba proba proba proba proba > proba5.txt`
- ▶ `cat proba5.txt`
- ▶ `more proba5.txt`
- ▶ `less proba5.txt`

počistite za sobom!

- ▶ `ls`
- ▶ `rm ls.ps`
- ▶ `ls`
- ▶ `rm -i ls.pdf`
- ▶ `ls`
- ▶ `^d`
- ▶ shutdown mašine, potom ugasite monitore
- ▶ vratite stolice na mesto
- ▶ ponesete sve svoje stvari

operativni sistem

— part 2 —

- ▶ **update**
- ▶ komandna linija, nastavak ... mnoštvo programa (jezik se zaboravlja ako se ne govori)
- ▶ scripting, bash, sh
- ▶ malo uvoda u Python, IDLE, idle
- ▶ malo C
- ▶ (ni)malo IDE, Integrated Development Environment, Code::Blocks, Eclipse (drugi to rade ...)

synaptic

- ▶ još jedan način za instaliranje programa
- ▶ jako dobro, omogućava precizna podešavanja
- ▶ nije instaliran by default ...
- ▶ zahteva neko razumevanje ...
- ▶ pokrenete Synaptic Package Manager
- ▶ dopunite octave, modul octave-control
- ▶ pokrenete **octave**
- ▶ `h = tf([1], [1, 1])`
- ▶ `bode(h)`
- ▶ `h = tf([10], [1, 0, 100])`
- ▶ `bode(h)`
- ▶ `h = tf([10], [1, 0.1, 100])`
- ▶ `bode(h)`

file system, nastavak ...

- ▶ bitno različit file system u odnosu na MS-DOS/win
- ▶ ne postoje volumes poput C:\, D:\, ...
- ▶ Ext 3, Ext 4, journal file system
- ▶ case sensitive, kada to može, ne kod FAT; ovo pravi probleme kod kopiranja na USB flash, File i file primer, preklapanje
- ▶ jedinstven file system, stablo počinje u root, /
- ▶ volumes se mount-uju: ubacite USB flash
- ▶ USB flash, CD, u /media/, pogledajte sa `ctrl+1`
- ▶ a sada kroz file system pomoću komandne linije ...
- ▶ terminal na pola, File Manager na pola, ...

directory system commands: pwd, mkdir, rmdir, cd ...

- ▶ `pwd` (koristite često da bi znali gde ste!)
- ▶ `mkdir <ime>`
- ▶ `ls`
- ▶ `cd <ime>`
- ▶ `pwd`
- ▶ `ls`
- ▶ `cd ..`
- ▶ `cd -`
- ▶ `cd ~`
- ▶ `cd /`
- ▶ `cd ~/<ime>`
- ▶ `cd`
- ▶ `cd <ime>`
- ▶ `pwd`

redirekcija, cat, tac, head, tail, more, less ...

- ▶ `ls -lht /usr/bin >usrbin1.lst`
- ▶ `ls /usr/bin >usrbin2.lst`
- ▶ `ls -lht >ls.lst`
- ▶ `ls`
- ▶ `cat usrbin1.lst`
- ▶ `tac usrbin1.lst`
- ▶ `head usrbin1.lst`
- ▶ `tail usrbin2.lst`
- ▶ `cat usrbin1.lst usrbin2.lst`
- ▶ `cat usrbin1.lst usrbin2.lst > usrbin3.lst`
- ▶ `more usrbin3.lst`
- ▶ `less usrbin3.lst`

pipe, grep ...

- ▶ `ls -lht /usr/bin | more`
- ▶ `ls -lht /usr/bin | less`
- ▶ `ls /usr/bin | more`
- ▶ `ls /usr/bin | less`
- ▶ `grep maxima usrbin1.lst`
- ▶ `grep maxima usrbin2.lst`
- ▶ `ls -lht /usr/bin | grep maxima`

grep i regular expressions

- ▶ `grep`, globally search a regular expression and print
- ▶ regular expression
- ▶ knjiga, 197 strana
- ▶ Google Analytics, Regular Expressions

scripting ...

- ▶ pokrenuti editor, `gedit radi`
- ▶ View, Highlight Mode, sh
↑ podešava context highlighting
- ▶ scripts su obično sa ekstenzijom sh
- ▶ mada ne mora ...
- ▶ videćete šta radi file

script file 2: briši

```
gedit brisi.sh  
  
rm -i lst/*.lst
```

privileges ...

- ▶ `ls -lhr`
- ▶ i postoji nešto nalik na `-rw-r--r--` pa podaci o fajlu
- ▶ i nešto nalik na `drwxr-xr-x` pa podaci o direktorijumu
- ▶ prvi znak: `-` za fajl ili `d` za direktorijum
- ▶ `r` je read, `w` je write, `x` je execute (ls)
- ▶ tripleti $\underbrace{rwx}_{\text{user}} \underbrace{rwx}_{\text{group}} \underbrace{rwx}_{\text{other}}$
- ▶ svaka privilegija (`r`, `w` ili `x` je jedan bit)
- ▶ triplet daje oktalanu cifru, `rw` je 7, `r--` je 4, `--x` je 1, ...
- ▶ privilegija se kodira kao trocifreni oktalni broj
- ▶ `rw`----- je 700
- ▶ `rw`xr--r-- je 744
- ▶ `rw`-r--r-- je 644

script file 3

```
gedit argumenti0.sh  
  
#!/bin/bash  
echo $0  
echo $1  
echo $2  
echo $3  
  
file argumenti0.sh  
sh argumenti0.sh  
sh argumenti0.sh a b c  
sh argumenti0.sh d e  
bash argumenti0.sh  
./argumenti0.sh
```

script file 1: radi

```
ls -lh /usr/bin >usrbin1.lst  
ls /usr/bin >usrbin2.lst  
ls -lh >ls.lst  
mkdir lst  
mv usrbin1.lst lst  
mv usrbin2.lst lst  
mv ls.lst lst  
cd lst  
ls -lh  
cd ..
```

startovanje shell file kao data file

- ▶ file radi, koji je tip fajla?
- ▶ `ls -lh radi`, biće još reči o ovome
- ▶ `bash radi`
- ▶ pogledajte šta je urađeno!
- ▶ `bash brisi.sh`
- ▶ pogledajte šta je urađeno!
- ▶ `sh radi`
- ▶ pogledajte šta je urađeno!
- ▶ `sh brisi.sh`
- ▶ pogledajte šta je urađeno!

chmod ...

- ▶ `ls -lh radi`
- ▶ `chmod <code> <file_name>`
- ▶ `chmod 700 radi`
- ▶ `ls -lh radi`
- ▶ pozeleneo?
- ▶ drugi način
- ▶ `chmod {u|g|o}{+|-}{r|w|x} <file_name>`
- ▶ `chmod u-w radi`
- ▶ `ls -lh radi`
- ▶ `chmod u+w radi`
- ▶ `ls -lh radi`
- ▶ `chmod go+x radi`
- ▶ `ls -lh radi`
- ▶ `chmod go-r radi`
- ▶ radi
- ▶ ./radi

#!

- ▶ `#!`
- ▶ promenite u `argumenti0.sh` prvi red u `#!/bin/sh`
i save kao `argumenti0a.sh`
- ▶ promenite u `argumenti0.sh` prvi red u `#!/usr/bin/python`
i save kao `argumenti0b.sh`
- ▶ obrišete prvi red i save kao `argumenti0c.sh`
- ▶ file `argumenti0.sh`
- ▶ file `argumenti0a.sh`
- ▶ file `argumenti0b.sh`
- ▶ file `argumenti0c.sh`

shell? malo istorije (za zainteresovane) ...

- ▶ https://en.wikipedia.org/wiki/Unix_shell
- ▶ https://en.wikipedia.org/wiki/Thompson_shell
- ▶ https://en.wikipedia.org/wiki/Bourne_shell
- ▶ https://en.wikipedia.org/wiki/Stephen_R._Bourne
- ▶ https://en.wikipedia.org/wiki/Bash_%28Unix_shell%29
- ▶ i jedna knjiga:
<http://sourceforge.net/projects/linuxcommand/>

script file 5

```
gedit argumenti2.sh
```

```
#!/bin/sh
echo 0: $0
echo 1: $1
echo 2: $2
echo 3: $3
```

```
file argumenti2.sh
sh argumenti2.sh
```

environment variables, PWD

- ▶ `pwd` (print working directory)
- ▶ `printenv PWD` (isto!)
- ▶ `echo $PWD` (isto!)
- ▶ `echo PWD` (ništa korisno)
- ▶ `$PWD` (pokuša da izvrši)
- ▶ `echo $pwd` (ništa)
- ▶ `echo $(pwd)` (ništa)
- ▶ `$(pwd)` (pokuša da izvrši)

čišćenje ... uvek!!!

- ▶ `cd lst`
- ▶ `ls`
- ▶ `ls -1`
- ▶ `rm -i *`
- ▶ `ls`
- ▶ `cd ..`
- ▶ `rmdir lst`

script file 4

```
gedit argumenti1.sh
```

```
#!/bin/bash
echo 0
echo $0
echo 1
echo $1
echo 2
echo $2
echo 3
echo $3
```

```
file argumenti1.sh
sh argumenti1.sh
./argumenti1.sh
```

environment variables, PATH

- ▶ `printenv`
- ▶ `printenv | less`
- ▶ `printenv PATH` (prikaže vrednost promenljive PATH)
- ▶ `printenv PATH | grep home`
- ▶ `echo $PATH` (prikaže vrednost promenljive PATH)
- ▶ `$PATH` (shell pokušava da izvrši)

proširimo PATH!

- ▶ `echo $PATH`
- ▶ `echo $PATH:$PWD`
- ▶ `PATH=$PATH:$PWD`
- ▶ `printenv PATH`
- ▶ `printenv PATH | grep home`
- ▶ `radi`

vaš prvi Python program, proba, bez .py ...

- ▶ python, imate li instaliran python koji nije python3?
- ▶ ako nemate, instalirajte, snađite se!
- ▶ za ovaj primer razlika je samo u ()

```
#!/usr/bin/python
print
print 'Hello World!'
print
print 'MPFC'
print
```

pokretanje, ...

- ▶ `python proba`
- ▶ `chmod +x proba`
- ▶ `./proba`
- ▶ `chmod -x proba`
- ▶ `./proba`
- ▶ `python proba`
- ▶ `cp proba proba.py`
- ▶ `ls -lhtr`
- ▶ `idle` (Not IDE, Eric Idle?)
- ▶ File, Open, proba.py
- ▶ context highlighting
- ▶ Run, Check Module (Alt+X)
- ▶ Run, F5

pokretanje ...

- ▶ da probamo sva tri nacina, sa i bez .py?
- ▶ `ipython`
- ▶ `run f.py`
- ▶ `ctrl/d` ili `^d`
- ▶ vrednosti promenljivih ostaju u interpreteru
- ▶ nema problema sa velikim brojevima, dynamic typing
- ▶ a kako je počelo? ovako:
- ▶ `python`
- ▶ `execfile('f.py')`
- ▶ `^d`

Fibonačijevi brojevi, Python

```
#!/usr/bin/python

n = input('duzina liste (int, >2): ')

print

a = b = 1

print 16 * '-'

i = 1
print repr(i).rjust(3) + ' ' + repr(a).rjust(10)
i = 2
print repr(i).rjust(3) + ' ' + repr(b).rjust(10)

for i in range(2,n):
    a, b = b, a + b
    print repr(i+1).rjust(3) + ' ' + repr(b).rjust(10)

print 16 * '-'
```

jedan C program ...

```
#include <stdio.h>

int main()
{
    int i, n, f;

    printf("\nargument = ");
    scanf("%d", &n);

    f = 1;
    for (i=1; i<=n; i++) {f=f*i;}

    printf("\n%d! = %d\n", n, f);

    return 0;
}
```

vaš drugi Python program, f.py, in gedit ...

```
#!/usr/bin/python

print
n = input('argument = ')
print

f = 1
for i in range(n):
    f = f * (i + 1)

print n, "! =", f
print
```

Zadatak: Fibonačijevi brojevi

- ▶ $f_n = f_{n-1} + f_{n-2}$, $f_1 = 1$, $f_2 = 1$
- ▶ lista od prvih n Fibonačijevih brojeva

zadatak, C program, faktorijel

- ▶ Dennis MacAlistair Ritchie (Username: dmr, September 8, 1941 – October 8, 2011)
- ▶ "UNIX is very simple, it just needs a genius to understand its simplicity."
- ▶ "C is quirky, flawed, and an enormous success."
- ▶ zatraži argument sa `argument =`
- ▶ izračuna faktorijel i javi rezultat u formi `m! = n`
- ▶ `gedit faktorijel.c`
- ▶ pišete program ...

GCC, prevođenje

- ▶ `gcc -o faktorijel faktorijel.c`
- ▶ `ls -lh`
- ▶ zeleni `faktorijel?`
- ▶ `./faktorijel`
- ▶ problemi sa velikim brojevima?

Fibonačijevi brojevi, C, simplified

```
#include <stdio.h>

int main(void)
{
    printf("\nn = ");
    int n;
    scanf("%d", &n);

    int i = 1, j = 1, k, c;
    printf("\n\n1\n1\n");

    for (c = 2; c <= n; c++)
    {
        k = i + j;
        i = j;
        j = k;
        printf("%d\n", k);
    }

    return 0;
}
```

korak napred, IDE, Code::Blocks

- ▶ IDE, Integrated Development Environment
- ▶ Eclipse, Code::Blocks i Geany popularni ... ovde C::B
- ▶ gdb, command line, sami
- ▶ kreirate projekat, Empty
- ▶ Debug/Release version
- ▶ Build
- ▶ watches
- ▶ breakpoints
- ▶ start, step into, shift-F7
- ▶ izvršavanje korak po korak, F7
- ▶ na kraju build Release version
- ▶ F7 ili button panel
- ▶ watch Local variables
- ▶ just to get acquainted ...

LaTeX

— I deo —

Uvod 1

- ▶ TeX (ili TeX), sistem za pripremu dokumenata, <http://en.wikipedia.org/wiki/TeX>
- ▶ Donald Knuth, http://en.wikipedia.org/wiki/Donald_Knuth, 70's, 1978 tačno
- ▶ Free/Open-Source Software (FOSS)
- ▶ standard za slaganje teksta, posebno naučnog, knjige, formule, import slika, čak i crtanje ...
- ▶ Computer Modern family of fonts, http://en.wikipedia.org/wiki/Computer_Modern
- ▶ može sve da se uradi, ali ponekad previše slobode škodi, ...

Uvod 2

- ▶ LaTeX (ili LaTeX), sistem makroa za TeX, <http://en.wikipedia.org/wiki/LaTeX>
- ▶ „a document markup language and document preparation system for the TeX typesetting program“
- ▶ Markup? Lako ćete naučiti **html** ako znate LaTeX!
- ▶ Leslie Lamport, http://en.wikipedia.org/wiki/Leslie_Lamport, 80's
- ▶ LaTeX je ono što se svakodnevno koristi
- ▶ ima još „nadogradnji“, LyX
- ▶ i dalje je sve free software
- ▶ i dalje svi dokumenti u ascii formatu, u slobodnom formatu, standardizovanom

Uvod 3

- ▶ suštinski, lako se uči
- ▶ zaista daje kvalitetan izgled dokumenata
- ▶ platforme na kojima je implementiran uglavnom free software
- ▶ sve što je potrebno postoji kao free software
- ▶ a potrebno je:
 - ▶ editor teksta
 - ▶ LaTeX procesor

Uvod 4, Leslie Lamport

- ▶ “How LaTeX changed the face of Mathematics”, an E-interview with Leslie Lamport, the author of LaTeX
- ▶ Q: *Is LaTeX hard to use?*
- ▶ A: *It's easy to use—if you're one of the 2% of the population who thinks logically and can read an instruction manual. The other 98% of the population would find it very hard or impossible to use.*

Uvod 5, Leslie Lamport

Even if there were some point to teaching kids such an esoteric system, it couldn't be done for the same reason that it's been impossible to raise the level of math and science in this country — namely, kids can't learn from teachers who don't know the subject well, and people who are good in math and science don't become grade-school teachers.

Uvod 6, Leslie Lamport

Q: *Three LaTeX mistakes that people should stop making?*

A:

1. *Worrying too much about formatting and not enough about content.*
2. *Worrying too much about formatting and not enough about content.*
3. *Worrying too much about formatting and not enough about content.*

Editori za LaTeX

- ▶ ima ih raznih, http://en.wikipedia.org/wiki/Comparison_of_TeX_editors
- ▶ neki free, neki ne
- ▶ koncentrišemo se na free, samo njih ćemo obraditi
- ▶ može bilo kakav editor, čak i gedit ima context highlighting za TeX, počecemo sa njim
- ▶ komforniji za rad su specijalizovani editori, nalik na IDE za programanje
- ▶ od specijalizovanih, **Texmaker**, moj izbor, jako napredovao, sada se ustalio, gotov proizvod
- ▶ Texmaker je „isti“ i pod win i pod lin (GNU/Linux)
- ▶ valja pomenuti i **TeXstudio**
- ▶ pod win je popularan i **TeXnicCenter**

LaTeX procesori

- ▶ pod lin TeX Live, uglavnom, ima još ...
- ▶ pod win uglavnom MikTeX ...
- ▶ prevodi *.tex file u *.dvi ili direktno u *.pdf
- ▶ pod lin, komandna linija

```
latex <fname>
```

ili

```
latex <fname>.tex
```

rezultat .dvi
- ▶ za direktno prevodjenje u *.pdf

```
pdflatex <fname>
```

ili

```
pdflatex <fname>.tex
```

LaTeX osobine

- ▶ obeležen tekst (markup language)
- ▶ obeležavanjem se označava šta sa pojedinim delovima teksta treba raditi, šta je naslov, šta je naslov poglavlja, šta je numerisana lista, šta je nenumerisana lista, šta je jednačina ...
- ▶ potrebno poznavati „komande“
- ▶ LaTeX „težak“ za „prosečnog“ korisnika
- ▶ WYSIWYM
- ▶ WYSIWYG (WYSIAYG) i WYSIWYM konvergiraju?
- ▶ preambula (header) i (obeležen) tekst

LaTeX resursi

- ▶ pod win:
 - ▶ <http://miktex.org/>
 - ▶ <http://www.ghostscript.com/> ili <http://pages.cs.wisc.edu/~ghost/>
 - ▶ možda <http://pages.cs.wisc.edu/~ghost/gsview/get49.htm>
 - ▶ neko (ja) više voli: <http://projects.gnome.org/evince/>
 - ▶ <http://www.texniccenter.org/>
 - ▶ ili (po mom ukusu): <http://www.xmlmath.net/texmaker/>

„A sada, svi na svoje radne zadatke!“

Ako hoćete da imate ikakve koristi od ovog časa:

1. unosite primere svojeručno
2. radite i koncentrišite se na to što radite
3. ako vas mrzi da kucate (!), imate na sajtu primere, kako source files tako i output files koje treba da dobijete
4. NIKAKO ne radite copy/paste sa pdf, biće problema sa ' i još nekim specijalnim znacima; kopirajte sa source files
5. zovite kada negde zapnete
6. ako vas sve ovo mrzi i dosadno vam je, **idite**.

LaTeX procesori

- ▶ dvi može da se gleda i štampa, http://en.wikipedia.org/wiki/Device_independent_file_format
- ▶ malo staromodan, mada radi ...
- ▶ može da se konvertuje u razne druge formate korišćenjem free software
 - ▶ dvips konvertuje dvi u ps
 - ▶ dvi2pdf, dvi2ps, dvi2psm konvertuju dvi u pdf
 - ▶ dvi2png konvertuje dvi u gif ili png
 - ▶ dvisvg, dvisvgm konvertuju dvi u svg
 - ▶ ...
- ▶ pravilo: što manje konverzija — to bolje

LaTeX resursi

- ▶ puno resursa na www, Google your problem
- ▶ <http://www.tug.org/>
- ▶ <http://www.tug.org/ctan.html>
- ▶ <http://www.andy-roberts.net/misc/latex/>
- ▶ <http://en.wikibooks.org/wiki/LaTeX>
- ▶ <http://upload.wikimedia.org/wikipedia/commons/2/2d/LaTeX.pdf>, pdf verzija, 748 strana

LaTeX resursi

- ▶ pod lin sve već imate u repository, dovućete Texmaker i to je to?
- ▶ ne bas, ponekad mora još nesto da se dodovuče, Synaptic
- ▶ bitno je da vam je mreža pri ruci ako nemate sredenu mašinu
- ▶ **po mom ukusu:** <http://en.wikibooks.org/wiki/LaTeX>, 748 strana
- ▶ **po mom ukusu:** <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>, 171 strana
- ▶ čitati uputstvo?
- ▶ ako baš mora, bolji hands-on approach!

primer 1, da počnemo

gedit, View, Highlight Mode, LaTeX

```
gedit p1.tex
```

```
\documentclass{minimal}
```

```
\begin{document}
```

```
Prva proba za \LaTeX.
```

```
\end{document}
```

procesiranje na komandnoj liniji

- ▶ u direktorijumu ste gde je p1.tex
- ▶ kucate `pdflatex p1` (ekstenzija .tex se ovde podrazumeva)
- ▶ sve je gotovo (ako nema poruka o greškama)
- ▶ pogledate pdf file sa `evince p1.pdf`
- ▶ ovaj prozor možete da ostavite, automatski refresh (pod lin, doduše)
- ▶ ovako ćemo raditi neko vreme, sticanja navike i stava radi

primer 3, dijakritici sa input encoding

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

\end{document}
```

značenje Tab i Enter

- ▶ Tab služi u formatiranju "source" teksta
- ▶ Tab se u izlazu ne vidi kao n*space
- ▶ zapravo, ne vidi se uopšte
- ▶ jedan Enter je separator kao space
- ▶ služi da se pregledno formatira "source"
- ▶ nov paragraf je prazan red, dva Enter
- ▶ **veoma** povoljno u praksi

primer 6, specijalni znaci ne rade

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

A ostaje i: \v{S}ta \'{c}e biti kada se pojave
na\v{s}i znaci?

A kada dodamo specijalne znake poput < i > ?

\end{document}
```

primer 2, dijakritici

```
\documentclass{minimal}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

\end{document}
```

primer 4, dijakritici bez input encoding

```
\documentclass{minimal}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

Mo\v{z}e ovako: \v{S}ta \'{c}e biti kada
se pojave na\v{s}i znaci?

\end{document}
```

primer 5, sređen input encoding

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

A ostaje i: \v{S}ta \'{c}e biti kada se
pojave na\v{s}i znaci?

\end{document}
```

primer 7, specijalni znaci rade, sređen font encoding

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

A ostaje i: \v{S}ta \'{c}e biti kada se pojave
na\v{s}i znaci?

A kada dodamo specijalne znake poput < i > ?

\end{document}
```

jedna posredna konverzija, za primer

```
latex p7
ls p7.dvi
dvips p7
ls p7.ps
ps2pdf p7.ps
ls p7.pdf
evince p7.pdf
```

može i `evince p7.ps` kao i `evince p7.dvi`

ovo se može automatizovati

```
gedit napravi
```

```
#!/bin/sh
latex $1
dvips $1
ps2pdf $1.ps
evince $1.pdf
```

```
file napravi
chmod u+x napravi
ls napravi
./napravi p7
PATH=$PATH:$PWD
napravi p7
```

Texmaker

- ▶ od sada idemo u prijatnije okruženje, Texmaker
- ▶ pogledati Tools
- ▶ PDFLaTeX je F6
- ▶ View PDF je F7
- ▶ LaTeX je F2
- ▶ Dvi -> PS je F4
- ▶ PS -> PDF je F8
- ▶ Dvi -> PDF je F9
- ▶ context highlighting, spell check, ...

primer 8, ћирилица у проблему

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

A ostaje i: \v{S}ta \'{c}e biti kada se pojave
na\v{s}i znaci?

A kada dodamo specijalne znake poput < i > ?

A како иде са Ћирилицом?

\end{document}
```

primer 9, ћирилица

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}

\begin{document}

Prva proba za \LaTeX.

Šta će biti kada se pojave naši znaci?

Ostaje i: \v{S}ta \'{c}e biti kada se pojave na\v{s}i znaci?

A kada dodamo specijalne znake poput < i > ?

A како иде са Ћирилицом?

\end{document}
```

primer 10, package datetime, today

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage{datetime}

\begin{document}

Ilustracija za packages:

\bigskip

Данас је \today.

\end{document}
```

primer 11, package babel

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\usepackage{datetime}

\begin{document}

Ilustracija za packages, sada je dodat babel, serbian:

\bigskip

Данас је \today.

\end{document}
```

primer 12, specijalni znaci

```
\begin{document}

A šta da radimo sa specijalnim znacima?

\bigskip

% ovo je komentar, neće se videti u dokumentu

ako treba u tekstu \% on se piše kao \textbackslash%.

можда вам затреба \textasciitilde % за ~

а можда и витичасте заграде \{ и \}

ima toga još: \$ \_ \& \pounds{} \copyright{}
\textless{} \textbar{} \textgreater{}

\end{document}
```

primer 13, okruženje verbatim

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Дођосмо до окружења. Прво радим окружење \verb-verbatim-

\begin{verbatim}

    Овде можете кучати шта год хоћете.
    И како год хоћете.
    Ништа се не форматира.
    Слободно кучаге \ ~ { }, нема никакво значење.

\end{verbatim}

\end{document}
```

primer 15, itemize u problemu

```
\documentclass{minimal}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Још мало окружења (environments). Листе су следећа тема.

\begin{itemize}
    \item прва тачка
    \item друга тачка
    \item трећа тачка
\end{itemize}

Ово баš и неће ваљати.

\end{document}
```

primer 17, enumerate

```
\documentclass{article}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Још мало окружења (environments).
Нумерисане листе су следећа тема.

\begin{enumerate}
    \item прва тачка
    \item друга тачка
    \item трећа тачка
\end{enumerate}

\end{document}
```

primer 19, margine i prazna strana

```
% margine na 2cm na a4 i da uklonimo broj strane

\documentclass[a4paper,12pt]{article}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[margin=2cm]{geometry}

\begin{document}

\pagestyle{empty} % ima i \thispagestyle{empty}

\Huge \noindent Али ово хоће! \\\
\normalsize Али ово хоће! \\\
\tiny Али ово хоће!

\end{document}
```

primer 14, noindent

```
\begin{document}

Још мало форматирања. Обрадили смо \verb+\bigskip+.

Да видимо шта ради \verb\noindent|.

\bigskip

\noindent Има \verb\noindent|.

Нема \verb\noindent|.

\noindent
Има \verb\noindent|.

Нема \verb\noindent|.

\end{document}
```

primer 16, itemize

```
\documentclass{article}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Још мало окружења (environments). Листе су следећа тема.

\begin{itemize}
    \item прва тачка
    \item друга тачка
    \item трећа тачка
\end{itemize}

Ово ће ваљати.

\end{document}
```

primer 18, veličina slova

```
\documentclass{article}

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}

\begin{document}

\noindent А сада мало играња са величином слова:

\Huge \noindent Али ово хоће! \\\
\huge Али ово хоће! \\\
\LARGE Али ово хоће! \\\
\Large Али ово хоће! \\\
\large Али ово хоће! \\\
\normalsize Али ово хоће! \\\
\small Али ово хоће! \\\
\footnotesize Али ово хоће! \\\
\scriptsize Али ово хоће! \\\
\tiny Али ово хоће!

\end{document}
```

primer 20, kvazitabela

```
\begin{document}

\thispagestyle{empty}

\noindent А сада једна табела:

\noindent
\Huge \verb\Huge| \\\
\huge \verb\huge| \\\
\LARGE \verb\LARGE| \\\
\Large \verb\Large| \\\
\large \verb\large| \\\
\normalsize \verb\normalsize| \\\
\small \verb\small| \\\
\footnotesize \verb\footnotesize| \\\
\scriptsize \verb\scriptsize| \\\
\tiny \verb\tiny| \\\

\end{document}
```

primer 21: još malo o fontovima i pravim tabelama

```
\begin{document}

\begin{tabular}{|l|l|}
\hline
команда & ефекат & \\
\hline
\verb|\textnormal| & \textnormal{Још мало о фонтовима (typefaces).} & \\
\verb|\textit| & \textit{Још мало о фонтовима (typefaces).} & \\
\verb|\underline| & \underline{Још мало о фонтовима (typefaces).} & \\
\verb|\textbf| & \textbf{Још мало о фонтовима (typefaces).} & \\
\verb|\emph| & \emph{Још мало о фонтовима (typefaces).} & \\
\verb|\textrm| & \textrm{Још мало о фонтовима (typefaces).} & \\
\verb|\textsf| & \textsf{Још мало о фонтовима (typefaces).} & \\
\verb|\textsc| & \textsc{Још мало о фонтовима (typefaces).} & \\
\verb|\textsl| & \textsl{Још мало о фонтовима (typefaces).} & \\
\verb|\texttt| & \texttt{Још мало о фонтовима (typefaces).} & \\
\hline
\end{tabular}

\end{document}
```

primer 22: gore, доле, лево, десно

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

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[margin=1cm]{geometry}

\begin{document}

\thispagestyle{empty}

горе лево \hfill горе десно

\vfill

доле лево \hfill доле десно

\end{document}
```

početak, zagrevanje

- ▶ update
- ▶ da napravimo jednu sliku ...
- ▶ `gnuplot` sa komandne linije
- ▶ `set terminal pdfcairo`
- ▶ `set output "slika.pdf"`
- ▶ `set output`
- ▶ `plot sin(x)`
- ▶ `set terminal epscairo`
- ▶ `set output "slika.ps"`
- ▶ `plot sin(x)`
- ▶ `set output`
- ▶ `Ctrl/d`
- ▶ pogledajte `slika.pdf` i `slika.ps`

struktura dokumenta, WYSIWYM

- ▶ zavisi od tipa dokumenta
- ▶ za `article`:
 - `\section{...}`
 - `\subsection{...}`
 - `\subsubsection{...}`
 - `\paragraph{...}`
 - `\subparagraph{...}`

primer a, struktura dokumenta, 1

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

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\usepackage{datetime}
\usepackage[bottom=3cm,top=3cm,left=3cm,right=2cm]{geometry}

\title{Primer jednog struktuiranog dokumenta}
\author{Predrag Pejović}
\date{\today}
```

primer a, struktura dokumenta, 2

```
\begin{document}

\maketitle

\section*{Abstract}
Ovo je samo primer za odeljak koji nije numerisan, \
* znači da se ne numeriše.

\section{Uvod}

U uvodu se piše šta se radi i zašto.
Važan deo uvoda je i pregled postojećih rezultata,
to pomaže da se objasni zašto se nešto radi.
```

primer a, struktura dokumenta, 3

```
\section{Centralni deo 1}

Ovde ilustrujemo subsections.

\subsection{Prvi pododeljak}
Kao što rekoh, prvi pododeljak.

\subsection{Drugi pododeljak}
Kao što rekoh, drugi pododeljak.

\section{Centralni deo 2}

Nastavljamo sa izlaganjem \ldots
```

primer a, struktura dokumenta, 4

```
\section{Zaključak}

U zaključku se rezimira šta je urađeno i kako.
Poneko voli i da kaže šta bi još moglo da se uradi.

\section*{Literatura}

Ovaj odeljak se obično ne numeriše,
ali referenciranje je baš duga priča.

\end{document}
```

šta je cilj primera?

- ▶ kako se automatski pravi naslov
- ▶ boldujte naslov, `\textbf{...}`, radi
- ▶ obrišite datum, `\date{}`
- ▶ struktura dokumenta, numerisanje odeljaka
- ▶ preglednost dokumenta, bilo gde nov red
- ▶ prazan red znači nov paragraf
- ▶ nov red sa `\newline` ili `\`, bilo ranije
- ▶ podelite dokument po stranama sa `\newpage` i/ili `\clearpage`
- ▶ vertical space, `\vspace{...}`

primer b, doterivanje naslova

```
\title{\LARGE  
\textbf{Primer jednog struktuiranog dokumenta}}  
\author{\textsl{Predrag Pejović}}  
\date{}
```

primer b, table of contents

iza naslova:

```
\vspace{3cm}  
\tableofcontents  
\newpage
```

nenumerisani items nisu u table of contents!

primer c, još malo environments, 1

```
\begin{document}  
  
\begin{Huge}  
  \begin{center}  
    Ovde su centrirana velika slova.  
  \end{center}  
\end{Huge}  
  
\begin{flushright}  
  ovde su normalna slova, \\  
  ali je sve poravnato \\  
  na desnu stranu  
\end{flushright}  
  
\end{document}
```

primer e, još malo environments, 3

```
\begin{document}  
  
\begin{description}  
  \item[Prvi] je onaj koji dolazi pre drugog  
  \item[Drugi] je onaj koji dolazi posle prvog  
  \item[Treći] je u ovom slučaju poslednji  
\end{description}  
  
\end{document}
```

primer b, brisanje broja strane

može (prva strana ili bilo koja specifična strana)

```
\thispagestyle{empty}
```

što je važno kada automatski generiše naslov

i/ili

```
\pagestyle{empty}
```

kad prođe naslov, odnosi se na strane osim prve

primer b, dodavanje items u table of contents

```
\newpage
```

```
\section*{Literatura}
```

```
\addcontentsline{toc}{section}{Literatura}
```

primer d, još malo environments, 2

```
\begin{document}  
  
\begin{flushleft}  
Ovaj environment (okruženje) ukida poravnavanje  
desne ivice, a to je ponekad potrebno.  
Primer su US patenti u postupku i neki pravni  
dokumenti za koje se veruje da su čitljiviji  
ako se ne poravnava desna ivica, mada manje  
lepo izgledaju. O ukusima ne vredi raspravljati,  
a u osnovi je sasvim svejedno.  
\end{flushleft}  
  
\end{document}
```

primer f, malo o razmacima

```
\documentclass[12pt, a4paper]{article}  
  
\usepackage[utf8]{inputenc}  
\usepackage[T1, T2A]{fontenc}  
  
\begin{document}  
  
Mi smo na ETF. Nismo na ETH.  
  
Mi smo na ETF\@. Nismo na ETH. % dodaje razmak  
  
\bigskip  
  
Ono je Prof. Davić.  
  
Ono je Prof.\ Davić. % smanjuje razmak  
  
\end{document}
```


primer g, još malo o razmacima

```
\documentclass[12pt, a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}

\begin{document}
\frechspacing

Mi smo na ETF. Nismo na ETH.

Mi smo na ETF\@. Nismo na ETH.

\bigskip

Ono je Prof. Davidć.

Ono je Prof.\ Davidć.

\end{document}
```

primer h, jednačine, superscript i grčka slova

```
Leonard Ojler je mnogo voleo jednačinu

\[ e^{-i \pi} + 1 = 0 \]

\noindent
Grčka slova su:  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\delta$ ,
\ldots  $\omega$ .
Velika grčka slova su:  $\Gamma$ ,  $\Delta$ , \ldots  $\Omega$ .
Svaki pristojan quick reference ima tabelu.
Većina editora, takode.
```

primer h, korenje i razlomci

```
Korenje:  $\sqrt{9}=3$ ,  $\sqrt[3]{8}=2$ .

\bigskip

Razlomci:  $\frac{1}{2}$ ,  $\displaystyle \frac{1}{2}$ ,
 $\frac{a}{b}$ ,  $\frac{a+c}{b}$ , može i  $1/2$ , a može i

\[ \frac{1}{2} \]

\[ \frac{a+b}{c} \]

Pogledajte package amsmath:  $\dfrac{}{}$  i  $\tfrac{}{}$ .
```

primer h, o zagradama

```
\noindent
Malo o zagradama:
\[ (\frac{a}{b}) \]
\[ \left( \frac{a}{b} \right) \]
\[ \big( \frac{a}{b} \big) \]
\[ \Big( \frac{a}{b} \Big) \]
\[ \bigg( \frac{a}{b} \bigg) \]
\[ \Bigg( \frac{a}{b} \Bigg) \]
I još malo:
\[ \left| x + j y \right| = \sqrt{x^2+y^2} \]
I još malo:
\[ k \in \left\{ 1, 2, \ldots n \right\} \]
```

primer h, jednačine, stilovi i subscript

```
\begin{document}

Numerisana jednačina:

\begin{equation}
i_{123} = i_1 + i_{2} + i_3
\end{equation}

\[ i_{123} = i_1 + i_{2} + i_3 \]

a može biti i u tekstu  $i_{123} = i_1 + i_{2} + i_3$ 

može i ovako  $\$i_{123} = i_1 + i_{2} + i_3 \$$ 
mada nije po standardu!

\end{document}
```

primer h, označavanje objekata

```
Neka je
\begin{equation}\label{prva}
e^{-i \pi} + 1 = 0
\end{equation}
Mi bi to pisali
\begin{equation}\label{druga}
e^{-j \pi} + 1 = 0
\end{equation}
Sada se treba pozivati na jednačinu (\ref{prva}) i
jednačinu (\ref{druga}).

Dobra praksa je da se u \label piše i tip objekta, da se ne pojave slika, tabela
i/ili jednačina sa istim imenom. Primer: \label{eq:prva}

Sređivanje referenci zahteva dva prolaza kompajlera, pokrenete ga dva puta za
redom.
```

primer h, integrali, razmaci i sume

```
Gustina verovatnoće:  $\int_{-\infty}^{\infty} p(x) \, dx = 1$ .
Ili  $\int \int_{-\infty}^{\infty} p(x) \, dx = 1$ .

Razmaci:

 $a$ ,  $b$ 

 $a$  \,  $b$ 

 $a$  \;  $b$ 

 $a$  \!  $b$ 

Jedno sumiranje:  $\sum_{k=1}^n k = \frac{n(n+1)}{2}$ . Ili
 $\sum_{k=1}^n k = \frac{n(n+1)}{2}$ 
```

primer h, funkcije, operatori

```
Funkcije se pišu u plain fontu (ima i priča oko razmaka), dosta ih je
uneto i počinju sa \. Neke nisu (npr. sgn)! Mogu se dodefinisati,
package amsmath, \operatorname.

\[ \sin(x+y)=\sin(x) \, , \cos(y)+\cos(x) \, , \sin(y) \]
\[ \sin^2 x + \cos^2 x = 1 \]
\[ \cosh^2 x - \sinh^2 x = 1 \]
\[ x \, , \operatorname{sgn}(x) = |x| \]
\[ \operatorname{h}(x)=1, \quad \text{if } x > 0 \]
\[ \operatorname{h}(x)=0, \quad \text{if } x < 0 \]
```

primer h, jednačine sa matricama

Matrice u jednačinama:

```
\[ \operatorname{h}(x) = \left\{ \begin{array}{l} 0, & \text{if } x < 0 \\ \frac{1}{2}, & \text{if } x = 0 \\ 1, & \text{if } x > 0 \end{array} \right. \]
```

primer h, jedinice mere, plain font, amsmath, text

```
\usepackage[greek, serbian]{babel}

\noindent Jedinice mere se pišu u plain fontu, \\\
razmahnute od „brojne vrednosti“.

\[ E = 10 \mbox{ V} \]
\[ E=10 \text{ V} \] % daje potpuno isti output
ispravno je:
\[ E = 10 \, \text{V} \]

\[ C = 100 \, \, \mbox{\textgreek{m}}\text{F} \]
\[ t_0 = 1\, \, \text{\textgreek{m}s} \]
%
Ovde ima problema sa grčkim slovima, mnogi rade:
%
\[ C=100 \, \, \mu\text{F} \] % namerno je oko = ovako
\[ t_0= 1 \, \, \mu\text{s} \] % namerno je oko = ovako
```

primer h, jedinice mere, upgreek

```
\usepackage{upgreek}

\[ C = 100 \, \, \mu\text{F} \]
\[ t_0=1 \, \, \mu\text{s} \]

\[ C=100 \, \, \mbox{\textgreek{m}}\text{F} \]
\[ t_0 = 1\, \, \text{\textgreek{m}s} \]

\[ C = 100 \, \, \upmu\text{F} \]
\[ t_0 = 1 \, \, \upmu\text{s} \]
```

primer h, matrice

```
Matrice:
\[ \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = a d - b c \]
```

Pogledajte još i okruženja: [matrix](#) i [pmatrix](#).

LaTeX
— III deo —
+ XCircuit
+ some scripting

priprema slika

- ▶ vektorska i rasterska grafika, Wikipedia
- ▶ za crteže prednost ima vektorska grafika
- ▶ formati od interesa dvi, ps, eps (za latex) i pdf (za pdflatex)
- ▶ od značaja još i png i jpg, mogu da se importuju u pdflatex
- ▶ konverzija programima dvips, ps2eps, ps2pdf, pstopdf i epstopdf, a ima i još
- ▶ nekih od ovih programa možda nema instaliranih na sistemu, instalirati
- ▶ od značaja:
 - ▶ **epstopdf**, konverzija eps slika u pdf
 - ▶ **pdfcrop**, obrezivanje pdf slika
 - ▶ **pdffonts**, prikazivanje fontova koji se koriste u dokumentu i njihovog statusa (embedded ili ne)
 - ▶ **convert**, konverzija formata slike
- ▶ pogledajte man za gornje programe

još malo o grafičkim formatima ...

- ▶ https://en.wikipedia.org/wiki/Comparison_of_graphics_file_formats
- ▶ https://en.wikipedia.org/wiki/Device_independent_file_format
- ▶ <https://en.wikipedia.org/wiki/PostScript>
- ▶ https://en.wikipedia.org/wiki/Encapsulated_PostScript
- ▶ https://en.wikipedia.org/wiki/Portable_Document_Format
- ▶ <https://en.wikipedia.org/wiki/JPEG>
- ▶ https://en.wikipedia.org/wiki/Portable_Network_Graphics
- ▶ <https://en.wikipedia.org/wiki/GIF>
- ▶ https://en.wikipedia.org/wiki/Scalable_Vector_Graphics
- ▶ sa gubicima (jpg), manji file, jača kompresija!
- ▶ bez gubitaka (png, gif)
- ▶ proprietary versus free

slike i konverzije

- ▶ napravili **sin.eps**
- ▶ probajte **ps2pdf sin.eps**
- ▶ pogledate sa **evince sin.pdf**
- ▶ baš ružno!
- ▶ da probamo **pdfcrop sin.pdf**
- ▶ **evince sin-crop.pdf**
- ▶ nešto je urađeno, ali ne valja, nema ticks
- ▶ **rm sin.pdf**
- ▶ **epstopdf sin.eps**
- ▶ **evince sin.pdf**
- ▶ to je to
- ▶ pokušaji nisu bili besmisleni, videćete smisao kod šema

komande za procesiranje i formati slika

- ▶ **F6**, ⇔ **pdflatex**
- ▶ **F7**, display pdf
- ▶ prvi prolaz, **??**, još nije pohvatao refs
- ▶ drugi prolaz, sada je sve ok, slika ispravno numerisana
- ▶ **F2**, tex → dvi, ⇔ **latex**
- ▶ **F4**, dvi → ps, ⇔ **dvips**
- ▶ **F8**, ps → pdf, ⇔ **ps2pdf**
- ▶ **F7**, display pdf
- ▶ latex "handle" samo ps i eps slike, konvertovati
- ▶ pdflatex "handle" pdf, png, jpg i gif, ne ps/eps
- ▶ probati

sin in gnuplot, po ko zna koji put, ako već nemate sliku ...

- ▶ komandna linija
- ▶ **gnuplot**
- ▶ **set terminal pdfcairo**
- ▶ **set output "slika.pdf"**
- ▶ **plot sin(x)**
- ▶ **set output**
- ▶ **set terminal epscairo**
- ▶ **set output "sin.eps"**
- ▶ **replot**
- ▶ **set output**
- ▶ **Ctrl/d**

primer, unošenje slika, pdflatex

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

\usepackage{graphicx} % package za unosenje slika
\usepackage[utf8]{inputenc}
\usepackage[T1,T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Ovde ćemo ubaciti jednu sliku, sliku \ref{slika:sinus}.

\begin{figure}[h!]
\centering
\includegraphics{sin.pdf}

\caption{Funkcija  $\sin \left( x \right)$ }
\label{slika:sinus}
\end{figure}

\end{document}
```

primer, unošenje slika, latex

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

\usepackage{graphicx} % package za unosenje slika
\usepackage[utf8]{inputenc}
\usepackage[T1,T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Ovde ćemo ubaciti jednu sliku, sliku \ref{slika:sinus}.

\begin{figure}[h!]
\centering
\includegraphics{sin.eps}

\caption{Funkcija  $\sin \left( x \right)$ }
\label{slika:sinus}
\end{figure}

\end{document}
```

primer, unošenje slika, scaling

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

\usepackage{graphicx} % package za unosenje slika
\usepackage[utf8]{inputenc}
\usepackage[T1,T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Ovde ćemo ubaciti jednu sliku, sliku \ref{slika:sinus}.

\begin{figure}[h!]
\centering
\includegraphics[scale=1.5]{sin.pdf}

\caption{Funkcija  $\sin$  \left( x \right)}
\label{slika:sinus}
\end{figure}

\end{document}
```

primer, unošenje slika, position

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

\usepackage{graphicx} % package za unosenje slika
\usepackage[utf8]{inputenc}
\usepackage[T1,T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Ovde ćemo ubaciti jednu sliku, sliku \ref{slika:sinus}.

\begin{figure} % [h!] % [h] % [t] % [t!] % [b!] % [b]
\centering
\includegraphics[scale=1.5]{sin.pdf}

\caption{Funkcija  $\sin$  \left( x \right)}
\label{slika:sinus}
\end{figure}

\end{document}
```

primer, unošenje tabela, centriranje

```
Ovde smo ubacili tabelu \ref{tabela:ime}.

\begin{table}[h!]
\centering
\caption{Primer jedne table} % ispod slike, iznad table

\begin{tabular}{r|c|l}
\hline
a & b & c \\
\hline
dddd & eeee & ffff \\
\hline
\end{tabular}
\label{tabela:ime}
\end{table}
```

table tricks

- ▶ još mnogo toga se može uraditi sa tabelama
- ▶ korisno: <http://www.tex.uniur.ac.ru/doc/tableTricks.pdf>
- ▶ nije loše imati gornji pdf kod sebe, 14 strana samo
- ▶ pitanja stila, obratite pažnju, kod nas sve table kao rešetke, pogledajte šta drugi rade, često samo tanke horizontalne linije

primer, unošenje slika, rotate

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

\usepackage{graphicx} % package za unosenje slika
\usepackage[utf8]{inputenc}
\usepackage[T1,T2A]{fontenc}
\usepackage[serbian]{babel}

\begin{document}

Ovde ćemo ubaciti jednu sliku, sliku \ref{slika:sinus}.

\begin{figure}[h!]
\centering
\includegraphics[angle=90, scale=1.5]{sin.pdf}

\caption{Funkcija  $\sin$  \left( x \right)}
\label{slika:sinus}
\end{figure}

\end{document}
```

primer, unošenje tabela

```
Ovde smo ubacili tabelu \ref{tabela:ime}.

\begin{table}[h!]
\centering
\caption{Primer jedne table} % ispod slike, iznad table

\begin{tabular}{|c|cc|}
\hline
a & b & c \\
\hline
d & e & f \\
\hline
\end{tabular}
\label{tabela:ime}
\end{table}
```

primer, unošenje tabela, phantoms

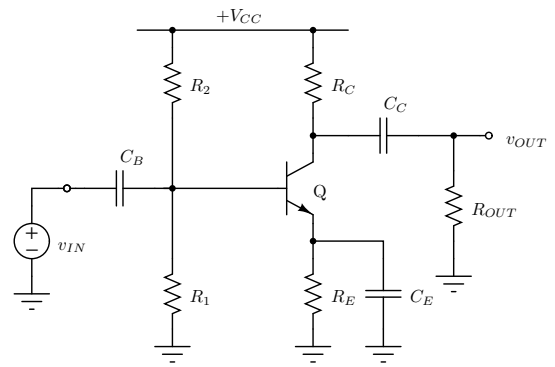
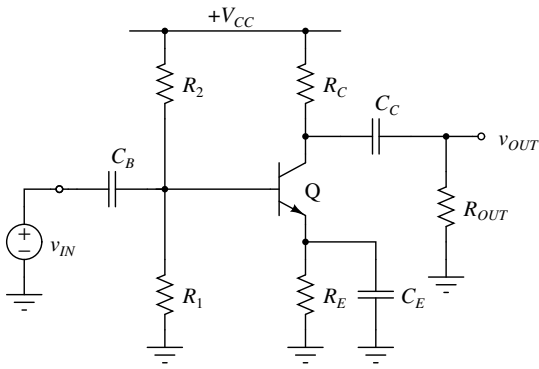
```
Ovde smo ubacili tabelu \ref{tabela:ime}.

\begin{table}[h!]
\centering
\caption{Primer jedne prazne table}
% caption ide ispod slike, iznad table

\begin{tabular}{|l|l|l|}
\hline
\phantom{XXXXX} & \phantom{XXXXX} & \phantom{XXXXX} \\
\hline
& \vphantom{\Huge{X}} & \\
\hline
\end{tabular}
\label{tabela:ime}
\end{table}
```

XCircuit

- ▶ program za crtanje električnih šema
- ▶ dominantan gde je od značaja izgled
- ▶ može da crta i lakše tehničke crteže
- ▶ malo neobičan interface
- ▶ malo neobična instalacija, synaptic potreban
- ▶ pravi PostScript files i sve radi na tom nivou
- ▶ konverzija u pdf sa **epstopdf**, fajl koji pravi XCircuit je .eps iako je ekstenzija .ps
- ▶ postoji i **ps2pdf**, razlike ...
- ▶ moguće je uneti i \LaTeX lettering
- ▶ pokreće se iz komandne linije sa **xcircuit**
- ▶ manual: <http://opencircuitdesign.com/xcircuit/>
- ▶ mora da se pokaže, da probate, nezgodno za opis rečima, ...
- ▶ radimo primer zajedno, posle vi sami, ja pomažem!



ime fajla: ce1.ps, convert to ce1.pdf, epstopdf ce1.ps

container file, ce2.tex

```
\documentclass[a4paper,12pt]{article}
\usepackage{graphicx}
\usepackage[margin=1cm]{geometry}
\pagestyle{empty}
\begin{document}
    \input{ce2-raw.tex}
\end{document}
```

procesiranje, varijanta 1

- ▶ latex ce2
- ▶ dvips ce2
- ▶ ps2eps -l -f ce2.ps
-l je za loose format, 1 pt margine,
-f je za force za overwrite
- ▶ ovde stajete ako koristite latex;
za pdflatex: epstopdf ce2.eps
- ▶ evince ce2.pdf ili evince ce2.eps

procesiranje, varijanta 2

- ▶ latex ce2
- ▶ dvips ce2
- ▶ do sada je isto kao pre, imamo ps file na celoj strani;
sada pocinju razlike zato da napravimo copy:
cp ce2.ps ce2-alt.ps
- ▶ ps2pdf ce2-alt.ps, pogledajte pdf
- ▶ pdfcrop ce2-alt
- ▶ evince ce2-alt-crop.pdf

procesiranje, script 1

```
napravite mf.sh

# make figure
latex $1
dvips $1
ps2eps -l -f $1.ps
epstopdf $1.eps

sh mf.sh ce2
```

generic container file, genfig.tex

```
\documentclass[a4paper,12pt]{article}
\usepackage{graphicx}
\usepackage[margin=1cm]{geometry}
\pagestyle{empty}
\begin{document}
    \input{\jobname-raw.tex}
\end{document}
```

procesiranje, script 2

```
napravite genmf.sh

# make figure, generic
cp genfig.tex $1.tex
latex $1
dvips $1
ps2eps -l -f $1.ps
epstopdf $1.eps

sh genmf.sh ce2
sh genmf.sh proba
```

procesiranje, script 3

napravite `genmf1.sh`, druga linija nema prekid, ima samo dve linije!

```
# make figure, generic
cp genfig.tex $1.tex && latex $1 && dvips $1 &&
ps2eps -l -f $1.ps && epstopdf $1.eps
```

```
sh genmf1.sh ce2
sh genmf1.sh proba
```

procesiranje, script 4

napravite `gmf.sh`, preglednost i estetika, primena \

```
# make figure, generic
cp genfig.tex $1.tex \
&& latex $1 \
&& dvips $1 \
&& ps2eps -l -f $1.ps \
&& epstopdf $1.eps
```

```
sh gmf.sh ce2
sh gmf.sh proba
```

procesiranje, script 5

napravite `gmfc.sh`, čišćenje za sobom

```
# make figure, cleanup
cp genfig.tex $1.tex \
&& latex $1 \
&& dvips $1 \
&& rm $1.tex \
&& rm $1.dvi \
&& rm $1.aux \
&& rm $1.log \
&& ps2eps -l -f $1.ps \
&& rm $1.ps \
&& epstopdf $1.eps \
&& rm $1.eps
```

```
sh gmfc.sh ce2
sh gmfc.sh proba
```

Referenciranje?

- ▶ pozivanje na prethodne publikacije, **važno!**
- ▶ praktično standardizovan format
- ▶ najčešće se lista referenci uređuje po redosledu pozivanja
- ▶ poziv na referencu je broj u uglastoj zagradi, npr. [1]
- ▶ u posebnom odeljku (section) „Literatura“ (“References”) mora biti navedeno šta je [1]
- ▶ potrebno je navesti **sve** podatke kako bi citirana referenca mogla biti jednoznačno određena
- ▶ često časopisi specificiraju strog format reference
- ▶ poseban format za knjige, radove u časopisima i radove na konferencijama

Referenciranje u \LaTeX -u

- ▶ \LaTeX ima sistem za automatsko numerisanje referenci, **izuzetno** važno u praksi, omogućava lake izmene dokumenta
- ▶ baza referenci važna stvar, \LaTeX ima nekoliko načina za unos referenci; ovde samo najjednostavniji
- ▶ za pravljenje baze referenci pogledajte package **BibTeX**, jako korisno ako pišete neki veći rad

standardna preambula

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

\usepackage[T1, T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[serbian]{babel}

\usepackage{graphicx}
\usepackage{amsmath}
\usepackage{xcolor}

\usepackage[margin=2cm]{geometry}

\title{Referenciranje}
\author{Predrag Pejović}
\date{\today}
```

tekst

```
\begin{document}

\maketitle

\section{Uvod}

Ovde ću citirati \cite{ND}.

\section{Zaključak}

Ovde citiram referencu reda radi, \cite{rr}.
```

reference

```
\begin{thebibliography}{[X]}

\bibitem{ND}
M.~Depenbrock, C.~Niermann,
“‘A new 12-pulse rectifier with
line-side interphase transformer
and nearly sinusoidal line currents,’”
\emph{Proceedings of the Power Electronics
and Motion Control Conference},
Budapest, 1990, pp.~374--378.

\bibitem{rr}
Ovo je referenca reda radi.

\end{thebibliography}

\end{document}
```

detalji, dashes

- ▶ <http://en.wikipedia.org/wiki/Dash>
- ▶ “hyphen”, -, push-pull
- ▶ “en-dash”, --, pp. 17–45
- ▶ “em-dash”, ---, IEEE Transactions on Circuits and Systems—Fundamental Theory and Applications
- ▶ “minus”, \$-\$, $3 - 2 = 1$
- ▶ da ih uporedimo: -, –, —, —
- ▶ vidite li razlike?

A sada nešto važno . . .

Beamer

beamer, uvod 1

- ▶ još jedna klasa dokumenata za \LaTeX
- ▶ popularan za pravljenje prezentacija
- ▶ rezultat je pdf file
- ▶ nema ništa od proprietary formata
- ▶ po svemu sudeći pdf će dugo moći da se čita, recycle stare prezentacije, ...
- ▶ Berkeley, EPFL, ...
- ▶ na „nemačkom“ je projektor beamer
- ▶ nije jedino rešenje, ali ubedljivo najzastupljenije

beamer, ciljevi

- ▶ osnovna prezentacija, frame (128mm x 96mm)
- ▶ dinamika prezentacije (pause, onslide)
- ▶ liste (itemized i enumerated)
- ▶ jednačine
- ▶ slike
- ▶ boje
- ▶ dve kolone
- ▶ handouts
- ▶ „teme“
- ▶ nema previše alata za sakrivanje nedostatka sadržaja u prezentaciji

napomena

```
%\end{frame}
```

znači:

```
\end{frame}
```

dok se ne reši bug!

beamer, resursi

- ▶ user guide, 247 strana,
<http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/doc/beameruserguide.pdf>
- ▶ primeri (jako važno!) http://www.informatik.uni-freiburg.de/~frank/ENG/latex-course/latex-course-3/latex-course-3_en.html
- ▶ preporučujem: <http://www.tug.org/pracjourn/2005-4/mertz/mertz.pdf>, 17 strana
- ▶ Q: Kako to obično funkcioniše?
- ▶ A: Primer koji vam se dopada dovučete i modifikujete.

beamer, header

- ▶ ne treba graphicx
- ▶ ne treba xcolor, color, ...

beamer, tipičan header

```
\documentclass{beamer}

% početak: može, ali ne mora
\usetheme{default} % {Copenhagen} {Goettingen}
\setbeamercovered{transparent} % ovo je važno za dinamiku
\usecolortheme{default} % {seahorse}{rose}{wolverine}{infolines}
% kraj: može, ali ne mora

\usepackage[utf8]{inputenc}
\usepackage[T1, T2A]{fontenc}
\usepackage[serbian]{babel}

\usepackage{datetime}

\title[prezentacija]{huge prezentacija}
\date{\today}
```

beamer, naslov, prvi slajd, doradivaćemo

```
\begin{document}

\begin{frame}

  \titlepage

%\end{frame}

\end{document}
```

dinamika običnog teksta

```
\section{dinamika običnog teksta}

\begin{frame}{jednostavna dinamika}

\pause
prva \\
\pause
druga \\
\pause
treća

%\end{frame}
```


složena dinamika

```
\begin{frame}{složena dinamika}

\onslide<2->{prva}\\
\onslide<3-3>{druga}\\
\onslide<2-2>{opet prva, kratko}\\
\onslide<4->{treća}\\

%\end{frame}
```

hide-unhide

```
\begin{frame}{hide-unhide}

\begin{itemize}
\item <2-2> prva stavka
\item <3-3> druga stavka
\item <4-4> treća stavka
\end{itemize}

%\end{frame}
```

numerisane liste

```
\section{enumerated list i jednačine}

\begin{frame}{numerisana lista}

\begin{enumerate}
\item <2-> prvi
\item <3-> drugi
\item <4-> treći
\end{enumerate}

%\end{frame}
```

jednačine

```
begin{frame}{malo matematike}

Pitagorina teorema glasi

\pause

\bigskip


$$c^2 = a^2 + b^2$$


\bigskip

\pause

znate li dokaz?

%\end{frame}
```

itemized list

```
\section{itemized list} % neće se odmah videti

\begin{frame}{nabrajanje}

\begin{itemize}
\item <2-> prva stavka
\item <3-> druga stavka
\item <4-> treća stavka
\item <2-> još jedna prva stavka
\end{itemize}

%\end{frame}
```

za lenje

```
\begin{frame}{za lenje}

\begin{itemize}
\item <+> jedan
\item <+> dva
\item <+> tri
\end{itemize}

%\end{frame}
```

progresivno boldovanje

```
\begin{frame}{progresivno boldovanje}

\begin{enumerate}
\item \textbf{<2->{prvi}}
\item \textbf{<3->{drugi}}
\item \textbf{<4->{treći}}
\end{enumerate}

%\end{frame}
```

potpuno sakrivanje

```
\begin{frame}{malo matematike, Pitagorin stil}

\setbeamercovered{invisible}
Pitagorina teorema glasi

\pause

\bigskip


$$c^2 = a^2 + b^2$$


\bigskip

\pause

znate li dokaz?
\setbeamercovered{dynamic} % ili {transparent}

%\end{frame}
```

detour: da napravimo sliku, ipython

komandna linija, terminal:

```
ipython --pylab
dm = 360
deg = linspace(- dm, dm, 2 * dm + 1)
x = radians(deg - 45)
y = cos(x)
plot(x, y)
xlim(- 2 * pi, 2 * pi)
ylim(- 1.5, 1.5)
xlabel('x')
ylabel('y')
```

save, `slika.png`

sakrivanje slike

```
\begin{frame}{slika}

\setbeamercovered{invisible}

\onslide<2->{često je potrebno ubaciti neku sliku}

\onslide<3->{\includegraphics[scale=0.3]{slika.png}}

\onslide<4->{što je odavno poznata funkcija}

\setbeamercovered{dynamic}

%\end{frame}
```

bojena podloga

```
\beamertemplatesolidbackgroundcolor{teal!20}

\begin{frame}{\textcolor{teal}{bojena pozadina}}

\begin{Huge}
\textcolor{teal}{sa tammim slovima}
\end{Huge}

%\end{frame}

\beamertemplatesolidbackgroundcolor{white}
```

handouts

```
\documentclass[handout]{beamer}
```

rešava posao, ali krupno

```
\usepackage{pgfpages}
\pgfpagesuselayout{2 on 1}[a4paper, border shrink=5mm]
\pgfpagesuselayout{4 on 1}[a4paper, landscape, border shrink=5mm]
\pgfpagesuselayout{8 on 1}[a4paper, border shrink=2mm]}
```

ima još puno toga, odlični web resursi!

ubacivanje slike

```
\section{mogu slike}

\begin{frame}{slika}

\pause

često je potrebno ubaciti neku sliku, npr. png sliku

\pause

\includegraphics[scale=0.35]{slika.png}

\pause

što je odavno poznata funkcija

%\end{frame}
```

bojena slova

```
\section{boje}

\begin{frame}{bojena slova}

\begin{Large}
\textcolor{red}{crvena} \\\
\pause
\textcolor{blue}{plava} \\\
\pause
\textcolor{cyan}{kako se ova zove?} \\\
\pause
\textcolor{magenta}{a ova?} \\\
\pause
\textcolor{yellow}{žuta} \\\
\pause
\textcolor{green}{zelena} \\\
\end{Large}

%\end{frame}
```

u dve kolone

```
\section{dve kolone}

\begin{frame}{dve kolone}

\begin{columns}[1]
\column{5cm}
\onslide <2-> gore levo \\\
\onslide <3-> dole levo \\\

\column{5cm}
\onslide <4-> gore desno \\\
\onslide <5-> dole desno \\\
\end{columns}

%\end{frame}
```

teme

- ▶ „teme“, stilovi, ...
- ▶ tema default na Goettingen
- ▶ sections sada dobijaju smisao
- ▶ tema na Copenhagen
- ▶ zašto li je ovo popularno?
- ▶ možete se igrati sa colortheme
- ▶ primeri: http://www.informatik.uni-freiburg.de/~frank/ENG/latex-course/latex-course-3/latex-course-3_en.html
- ▶ vodite računa o vremenu!
- ▶ ovo će biti deo (20%) drugog kolokvijuma!

GNU Octave

- ▶ vrlo često u tehni
- ▶ retko stvarni problemi imaju closed-form solution
- ▶ čak i kad imaju, pitanje upotrebljivosti
- ▶ mnogo detalja — numerički pristup
- ▶ u početku tretirano kao numerologija ...
- ▶ kolika je „saznajna vrednost“ nekog rešenja i kako se meri?

Numerical Computation, Tools #1: Matlab

- ▶ diminira(o) Matlab
- ▶ Wikipedia, http://en.wikipedia.org/wiki/Comparison_of_numerical_analysis_software, 16.12.2018:
 - ▶ “Numerical computation and simulation with extended 2D/3D visualization.”
 - ▶ “Proprietary”
 - ▶ 2014: “\$2150 (commercial) / \$99 (student)”
 - ▶ 2014: \$300 manje nego pre tri godine, isto kao prošle i prethodne godine
 - ▶ 2015: “\$2650 (standard) / \$625 (education)”
 - ▶ 2017: “\$2650 (standard) / \$625 (education)”
 - ▶ 2018: “\$2150 (standard) / \$500 (education) / \$149 (home) / \$49 (student)”
- ▶ osnovni tip podatka: n-dimenziona matrica
- ▶ extended 2D/3D visualization ...

red je da pomenemo (i ulinkujemo) i neke alternative ...

- ▶ Scilab, još jedan Matlab-like, French ...
- ▶ FreeMat (prilično neaktivan poslednjih godina)
- ▶ Python, PyLab ... prednosti modernog jezika
- ▶ R, statistika
- ▶ Julia, jako brzo, vrlo budućnost ...
- ▶ ...

literatura, dokumentacija

- ▶ dokumentacija koja ide uz GNU Octave, odlična, tu je sve što je realno potrebno
- ▶ <https://www.gnu.org/software/octave/octave.pdf>, 1004 strane!
- ▶ tutorials na www, mnogo
- ▶ izbor linkova, <http://tnt.etf.rs/~oe4sae/>
- ▶ napomena: GNU Octave 4.4.1, 1060 strane!!!
- ▶ GNU Octave 4.4.1, August 2018
- ▶ terminal, `octave-cli`, default je sada GUI!
- ▶ terminal, `octave --force-gui` za ver < 4.0.0 ili `octave` za ver >= 4.0.0
- ▶ `ver`

Numerical Computation, Tools #2: GNU Octave

- ▶ GNU Octave
 - ▶ “General numerical computing package with lots of extension modules. Syntax mostly compatible with MATLAB”
 - ▶ “GPL”
 - ▶ “Free”
- ▶ <https://www.gnu.org/software/octave/>

Odakle se nabavlja GNU Octave?

- ▶ GNU/Linux, Ubuntu:
 - ▶ repository, Ubuntu Software Center
 - ▶ GNU Octave
 - ▶ Synaptic Package Manager
 - ▶ search “octave”
 - ▶ toolboxes, dodatna dokumentacija, ...
- ▶ win:
 - ▶ nije bio kao pod GNU/Linux
 - ▶ ima toolboxes
 - ▶ sada jedino MXE builds: <https://ftp.gnu.org/gnu/octave/windows/> prvo pokretanje grafike može dugo da traje, posle ok!
- ▶ Da li vam je potreban GUI? Ima sada i to, by default!

pocinjemo, GNU Octave kao kalkulator

```

2+2
2 + 2 # može space oko, preporuka
2-3
2*3
3/4
3.0/4.0
10/3
-10/3
2^3
3^2
3^3
10^10 % sve su to floats!
3^64
3^640 # još nije Inf
    
```

Inf i NaN

```
3^6400 # Infinity, Inf
5 / Inf
5 * Inf
0 / 3
3 / 0
0 / 0 # Not a Number, NaN
4 * NaN
5 / NaN
```

promenljive, ; i ans

```
a = 3
a = 4; # odziv suppressed
a
disp(a) # uočite razliku, nema a = . . .
b = 5
a + b
ans # kod Python-a je ovo _
```

strukture podataka

```
tacka1.x = 5
tacka1.y = 7
tacka2.x = 1
tacka2.y = -1
tacka1 + tacka2
tacka1.x + tacka2.x
# slab overloading
```

stringovi

```
a = 'Pera'
b = "Mika"
a + b # čudan string?
strcat(a, b)
a == b
a(1)
a(3)
b(1:3)
b(1 : 3)
b (1 : 3)
```

help!

```
help(floor)
help floor # short description
doc floor # documentation, long description
# izlazak sa q
help ceil
doc ceil
# izlazak sa q
help
doc
```

relacioni operatori

```
2 < 3
3 < 2
2 <= 2
2 >= 3
2 ~= 2
2 != 2 % malo proširenje u odnosu na Matlab
2 == 2 % pazite se, često pravi bugs!!!
```

logički operatori

```
1 & 0
1 & 1
0 & 0
0 | 0
1 | 0
1 | 1
~1
!1
~0
!0
```

logički operatori, nad nizovima

```
a = [0 1 0 1]
b = [0, 0, 1, 1]
a & b
a | b
!a
~b
a' & b'
a' | b'
[a' !a']
[a' b' a' & b']
[a' b' a' | b']
```

operatori inkrementiranja

```
c = 0
++c
c++
c
c--
c
--c
c
```

matrice, indeksiranje i transpose, 1

```
a = 1 : 5
a(1) # razlika u odnosu na Python!!!
a(5)
a(2 : 4)
a(-1)
length(a)
size(a)
b = a'
b(4)
length(b)
size(b)
```

neke posebne matrice

```
eye(3)
zeros(4)
zeros(4, 1)
zeros(1, 4)
ones(5)
ones(1, 5)
```

scripts, Gauss na loš način, run

```
ls
ls gauss.m
help gauss
gauss
100
```

matrice

```
a = [1, 2; 3, 4]
a = [1 2; 3 4] # ovo ne može u Python-u
a = [1 2
3 4] # ni ovo

b = inv(a)
a * b
b * a
a .* b
b .* a
a ./ a
a / a
inv(a) * a
a / b
a * a
```

matrice, indeksiranje i transpose, 2

```
c = [1 2 3; 4 5 6]
length(c)
size(c, 1)
size(c, 2)
c(2, 2)
c(2, :)
c(:, 2)
c(2 : 4) # pazite se, bugs!!!
c
```

scripts, Gauss na loš način

```
gedit gauss.m
```

View / Highlight Mode / Scientific / Octave

```
# Gauss, na los nacín

n = input('do koliko? ');

s = 0;

for i = 1 : n
    s += i;
end

printf('sum = %d\n', s)
```

moglo je i ovako

```
a = 1 : 100 # ili a = 1 : 100;
sum(a)
# ili
a = ones(1, 100)
b = cumsum(a)
sum(b)
mean(a)
mean(b)
```

save, prvi deo

```
clear all
a = 3
b = 5
string = 'string'
x = [1 2; 3 4]
save ws # ceo workspace!
```

a sada uradite

```
gedit ws
```

load, prvi deo

```
a
b
string
clear all
a
b
string
load ws
a
b
string
```

funkcije sa nizovima

```
deg = 0 : 360;
length(deg)
wt = deg * pi / 180;
s = sin(wt);
s2 = s .* s;
rmss = sqrt(mean(s2))
rmss - 1/sqrt(2)
c = cos(wt);
c2 = c .^ 2;
rmsc = sqrt(mean(c2(1 : length(c2) - 1)))
rmsc - 1/sqrt(2)
```

plot, 2d

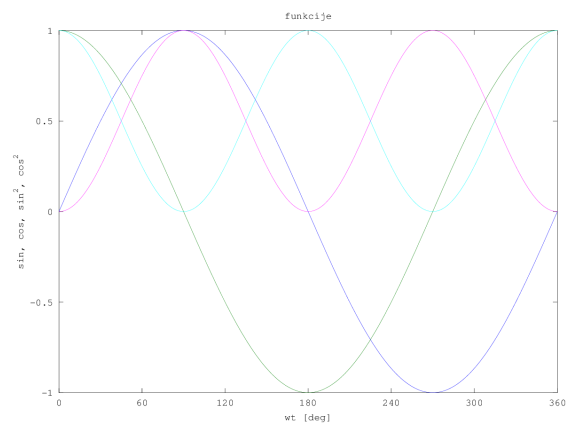
```
plot(deg, s)
plot(deg, c)
hold on
plot(deg, s, 'r')
close all
plot(deg, s, deg, c)
hold on
plot(deg, s2, 'm', deg, c2, 'c')
axis([0 360 -1 1]) # mora , in Python
set(gca, 'XTick', 0 : 60 : 360)
xlabel('wt [deg]')
ylabel('sin, cos, sin^2, cos^2')
title('funkcije')
```

plot, 2d

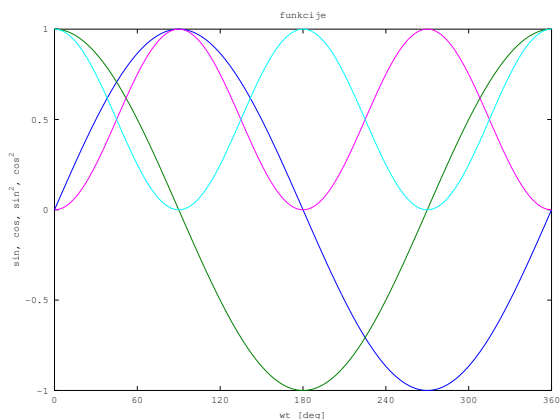
```
help figure
doc figure
help print
doc print
print('slika.png', '-dpng')
print('slika.pdf', '-dpdf')
print('slika.eps', '-deps')
```

pogledajte png, pdf, eps sliku, različite u odnosu na ekran!
svaki drajver specifičan, kao gnuplot (što i jeste)

slika.png



slika.pdf, cropped



save i load, drugi deo

```
data = [deg' wt' s' c' s2' c2'];
data
size(data)

save data data

clear all

data

load data

data

gedit data
```

hoću da rešim sistem jednačina:

$$x + y = 3$$

$$x - y = 1$$

```
a = [1, 1; 1, -1]
```

```
b = [3; 1]
```

```
b = [3 1]'
```

```
a \ b # linear least squares
```

```
inv(a) * b
```

```
cond(a) # condition number
```

```
rcond(a)
```

```
rref(a) # reduced row echelon form
```

- ▶ GNU Octave je jako bogat funkcijama!
- ▶ Sjajan reference manual, 1060 strana!
- ▶ Ovo uputstvo će biti dopunjavano, proširivano ...
- ▶ Bar je takav **bio** plan ...
- ▶ A sada? Koristi se, ali Python, Julia, ...
- ▶ Mora da se predaje, Matlab je uzrok.
- ▶ Mnogo toga ima, prioritet?
- ▶ Od kada je ovaj predmet započet, 2010. godine, mnogo toga je nastalo!

gnuplot



gnuplot ...

- ▶ dugovečan program (1986) za vizuelizaciju podataka, stalno dograđivan
- ▶ ugrađen u GNU Octave, Maxima, R, ...
- ▶ crta podatke (iz fajla) i funkcije (analitički zadate)
- ▶ 2D i 3D, eksplicitno i parametarski
- ▶ Dekartov i polarni koordinatni sistem
- ▶ komandna linija, danas za većinu korisnika problem
- ▶ veoma kvalitetan izlaz
- ▶ razvija se: nove verzije, knjige ...
- ▶ mnogima težak, mada je sve to ista stvar ...
- ▶ vole ga statističari i zainteresovani za advanced data analysis

... gnuplot ...

- ▶ glavni izvor: <http://www.gnuplot.info/>
- ▶ sada aktuelna verzija 5.2.2
- ▶ za win sada instalacija, ...
- ▶ uputstvo
http://www.gnuplot.info/docs_5.2/Gnuplot_5.2.pdf,
271 strana
- ▶ pomoćni izvor: <http://www.gnuplotting.org/>
- ▶ proprietary:
 - ▶ Gnuplot in Action
 - ▶ gnuplot Cookbook
 - ▶ gnuplot 5
- ▶ očigledno se pišu knjige na gnuplot temu ...

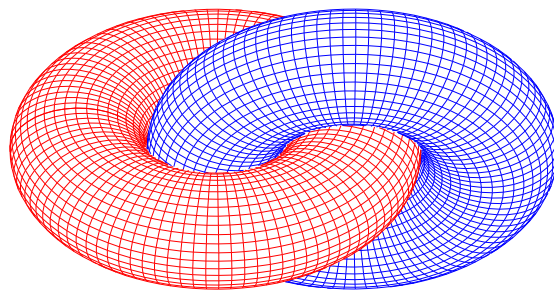
... gnuplot

- ▶ program stabilan, sa mnogo korisnika, ima smisla učiti ga
- ▶ veoma puno resursa dostupno
- ▶ sjajna zbirka primera <http://gnuplot.sourceforge.net/demo/>
- ▶ odaberete šta vam treba, preradite, brzo i efikasno, moj slučaj, Madrid ...
- ▶ alternative:
 - ▶ Origin, US\$ 1095 (std), US\$ 1800 (pro), US\$ 850 (academic pro), US\$ 69/yr (student)
 - ▶ SigmaPlot, sa njihovog sajta: Academic: \$599, Commercial: \$899, Government: \$799
- ▶ imate neke primedbe na komandnu liniju?

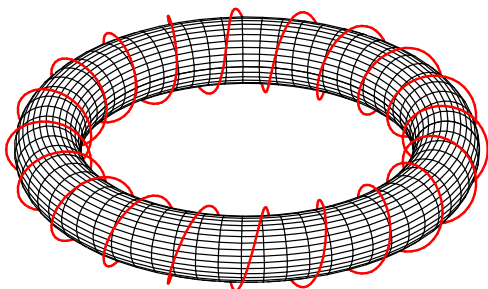
gnuplot, alternative ...

- ▶ jedna od GUI free alternativa SciDAVis (Python)
- ▶ jos poznati LabPlot, Grace, Vezsz (Python)
- ▶ PyXPlot, vrlo slična sintaksa (Python)
- ▶ ...
- ▶ treba li vam išta od toga?
- ▶ ja mislim: **ne**
- ▶ lično iskustvo: što standardnije, to bolje
- ▶ lično iskustvo: što veća baza korisnika, to bolje
- ▶ slike se recikliraju ...

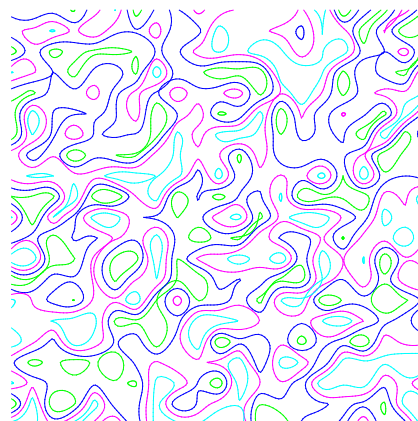
gnuplot, za šta je dobar?



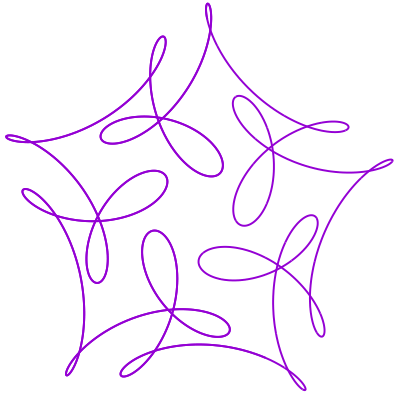
gnuplot, za šta je dobar?



gnuplot, za šta je dobar?



gnuplot, za šta je dobar?



gnuplot, za šta je dobar?



gnuplot, xrange, yrange

- ▶ `set xrange [-5*pi:5*pi]`
- ▶ `show xrange`
- ▶ `replot`
- ▶ `set yrange [-1.1:1.1]`
- ▶ `show yrange`
- ▶ `replot`
- ▶ `reset`
- ▶ `replot`

gnuplot, more functions, key

- ▶ `plot sin(x)/x, -sin(x)/x`
- ▶ `unset key`
- ▶ `replot`
- ▶ `set key left top`
- ▶ `replot`
- ▶ `set key right bottom box`
- ▶ `replot`
- ▶ `help set key`
- ▶ `set key tmargin left (ostaje box!)`
- ▶ `replot`
- ▶ `set key rmargin center nobox`
- ▶ `replot`
- ▶ `set key above box; replot`
- ▶ `set key below nobox; replot`
- ▶ ima još opcija, help, books, ...

gnuplot, za šta je dobar?



gnuplot, terminal -> terminal

- ▶ idemo na komandnu liniju, crveno su gnuplot komande koje kucate
- ▶ `gnuplot`
- ▶ `help`
- ▶ `plot sin(x)/x`
- ▶ `print pi`
- ▶ `print 5*pi`
- ▶ `print 2**3`
- ▶ ima i promenljive
- ▶ `a=pi/4`
- ▶ `print sin(a)**2`

gnuplot, samples

- ▶ `show samples`
- ▶ `set samples 10`
- ▶ `plot sin(x), cos(x)`
- ▶ `show samples`
- ▶ `set samples 360*5`
- ▶ `replot`
- ▶ `show samples`
- ▶ `help set samples`

gnuplot, tics

- ▶ `plot [-2*pi:2*pi] sin(x), sin(x)**2`
- ▶ `set xtics (-2*pi, -pi, 0, pi, 2*pi); replot`
- ▶ `set xtics nomirror; replot`
- ▶ `set grid; replot`
- ▶ `set xtics (-2*pi, -3*pi/2, -pi, -pi/2, 0, pi, 2*pi); replot`
- ▶ `unset grid; replot (nomirror ostaje!)`
- ▶ `set xtics ("-pi" -pi, "0" 0, "pi" pi); replot`
- ▶ `reset; replot`
- ▶ `set xtics pi`
- ▶ `set mxtics 2; replot`
- ▶ `set grid mxtics; replot`
- ▶ `set grid xtics mxtics ytics; replot`

gnuplot, lines and points

- ▶ `reset; plot sin(x), cos(x)`
- ▶ `plot sin(x) with lines linetype 1 linewidth 1, cos(x) with lines linetype 1 linewidth 2`
- ▶ `plot sin(x) with lines linetype 3 linewidth 2, cos(x) with lines linetype 4 linewidth 1`
- ▶ `plot sin(x) with points`
- ▶ `plot sin(x) with linespoints`
- ▶ `plot sin(x) with dots`
- ▶ `plot sin(x) with points, cos(x) with points`
- ▶ `set samples 10`
- ▶ `set yrange [0:6]`
- ▶ `plot 1 with points pointtype 1, \`
`2 with points pointtype 2, \`
`3 with points pointtype 3, \`
`4 with points pointtype 4`

gnuplot, labels

- ▶ `plot sin(x)`
- ▶ `set xlabel "x"`
- ▶ `set ylabel "y"`
- ▶ `set title "y=sin(x)"`
- ▶ `replot`
- ▶ `set label "neki tekst" at 5,0.5; replot`
- ▶ `plot sin(x) title "sin", cos(x) title "cos"`
- ▶ `set key box; replot`

gnuplot, terminals and output 2

- ▶ `set output`
- ▶ `show output`
- ▶ `show terminal`
- ▶ menjate terminal koji imate, qt ili wxt, ja pisem za wxt, važi i na dalje
- ▶ `set terminal wxt 0 persist`
- ▶ `plot sin(x)`
- ▶ `set terminal wxt 1 persist`
- ▶ `plot cos(x)`
- ▶ `set terminal wxt 0`
- ▶ `show terminal`
- ▶ `reset`

gnuplot, file data plotting

- ▶ `plot "dat" using 1:3`
- ▶ `plot "dat" using 1:3 with lines`
- ▶ `plot "dat" using 2:3 with lines`
- ▶ `plot "dat" using 1:3 with lines, "dat" using 1:4 with lines, "dat" using 1:5 with lines`
- ▶ `plot "dat" using 1:3 with lines linewidth 2, \`
`"dat" using 1:4 with lines linetype 2, \`
`"dat" using 1:5 with lines linetype 2`
- ▶ `plot "dat" using 1:3 with lines linewidth 2 title "Vc", \`
`"dat" using 1:4 with lines linetype 2 title "anv 1", \`
`"dat" using 1:5 with lines linetype 2 title "anv 2"`
- ▶ `set xrange [0:360*4]`
- ▶ `set xtics 180; replot`
- ▶ postaje zamorno; vreme je za scripting

gnuplot, steps and bars

- ▶ `reset; plot sin(x) with impulses`
- ▶ `plot sin(x) with steps`
- ▶ `plot sin(x) with steps, sin(x) with fsteps, sin(x) with histeps`
- ▶ `plot sin(x) with boxes`
- ▶ `set style fill solid 1; replot`
- ▶ `set boxwidth 0.5 relative; replot`

gnuplot, terminals and output 1

- ▶ `show terminal`
- ▶ `help terminal`
- ▶ `help terminal postscript`
- ▶ `set terminal postscript eps`
- ▶ `show output`
- ▶ `set output "slika1.ps"`
- ▶ `plot sin(x), cos(x)`
- ▶ pogledajte šta ste dobili kao slika1.ps
- ▶ `set output "slika2.ps"`
- ▶ `set terminal postscript eps color`
- ▶ `plot sin(x), cos(x)`
- ▶ pogledajte šta ste dobili kao slika2.ps

gnuplot, da napravimo file

- ▶ `octave`
- ▶ `deg=0:360*4;`
- ▶ `wt=deg*pi/180;`
- ▶ `f=exp(-wt/2/pi*0.5).*sin(wt);`
- ▶ `plot(deg,f)`
- ▶ `fp=exp(-wt/2/pi*0.5);`
- ▶ `fm=-fp;`
- ▶ `dat=[deg' wt' f' fp' fm'];`
- ▶ `save dat dat`
- ▶ `exit` ili `ctrl-d` (`^D`)
- ▶ `gedit dat`
- ▶ `# znači komentar`
- ▶ zapamtiti šta je u kojoj koloni, 1, 2, 3, 4 i 5

gnuplot, prvi script

- ▶ da napravimo script, `gedit slika.gp`
- ▶ ne mora ekstenzija `.gp`, moze bilo šta
- ▶ kucate:

```
set xrange [0:360*4]
set xtics 180
set xlabel "wt [deg]"
set ylabel "Vc"
plot "dat" using 1:3 with lines linewidth 2 title "Vc", \
    "dat" using 1:4 with lines linetype 2 title "anv 1", \
    "dat" using 1:5 with lines linetype 2 title "anv 2"

gnuplot
load "slika.gp"
```

gnuplot, još terminala

- ▶ `set terminal png`
- ▶ `set output "slika3.png"`
- ▶ `replot`
- ▶ `set terminal pdfcairo`
- ▶ `set output "slika3.pdf"`
- ▶ `replot`
- ▶ `set terminal jpeg`
- ▶ `set output "slika3.jpg"`
- ▶ `replot`
- ▶ `set terminal wxt`
- ▶ `set output`
- ▶ pogledajte `slika3.png`, `slika3.pdf` i `slika3.jpg`; razlike?

gnuplot, latex terminal, cover

- ▶ napravite tex cover `slika4.tex`:

```
\documentclass{minimal}

\begin{document}

    \input{s14}

\end{document}
```
- ▶ može Texmaker, a može i komandna linija `pdflatex slika4`, direktno u pdf
- ▶ može i indirektno, `dvi-ps-pdf`: `latex slika4`, `dvips slika4`, `ps2pdf slika4.ps`
- ▶ može da se „kropuje“, `ps2eps -l slika4.ps`, `epstopdf slika4.eps`
- ▶ može i `latex slika4`, `dvipdf slika4`, nema kropovanja

gnuplot, epslatex color terminal

- ▶ napravite script

```
set terminal epslatex standalone color
set output "slika6.tex"
set xrange [0:360*4]
set xtics 180
set xlabel "$\omega t \;; [-\circ]$"
set ylabel "$v_C$"
plot "dat" using 1:3 with lines linewidth 2 title "$v_C$", \
"dat" using 1:4 with lines linetype 2 title "anv$_1$", \
"dat" using 1:5 with lines linetype 2 title "anv$_2$"
```
- ▶ procesirate kroz gnuplot, dobili ste standalone tex
- ▶ procesirate kroz latex, procesirate preko `dvi` i `ps` do pdf
- ▶ pogledate i uporedite sve slike od sada

gnuplot, another parametric plot

create `mystery-curve.gp` as follows:

```
set terminal epslatex standalone color size 6, 6
set output "mystery-curve.tex"
```

```
set parametric
set samples 10000
set size square
```

```
set key off
unset xtics
unset ytics
unset border
```

```
plot cos(t)+cos(6*t)/2+sin(14*t)/3, \
sin(t)+sin(6*t)/2+cos(14*t)/3 linewidth 4
```

gnuplot, latex terminal

- ▶ napravite script `s14.gp`

```
set terminal latex
set output "s14.tex"
set xrange [0:360*4]
set xtics 180
set xlabel "$\omega t \;; [-\circ]$"
set ylabel "$v_C$"
plot "dat" using 1:3 with lines linewidth 2 title "$v_C$", \
"dat" using 1:4 with lines linetype 2 title "anv$_1$", \
"dat" using 1:5 with lines linetype 2 title "anv$_2$"
```
- ▶ mora `\` umesto `\` pošto je `\` gnuplot znak za novi red
- ▶ može `load "s14.gp"`
- ▶ a može i sa komandne linije: `gnuplot s14.gp`
- ▶ dobili ste `s14.tex` koji nije standalone

gnuplot, epslatex terminal

- ▶ napravite script

```
set terminal epslatex standalone
set output "slika5.tex"
set xrange [0:360*4]
set xtics 180
set xlabel "$\omega t \;; [-\circ]$"
set ylabel "$v_C$"
plot "dat" using 1:3 with lines linewidth 2 title "$v_C$", \
"dat" using 1:4 with lines linetype 2 title "anv$_1$", \
"dat" using 1:5 with lines linetype 2 title "anv$_2$"
```
- ▶ procesirate kroz gnuplot, dobili ste standalone tex
- ▶ latex `slika5`, ne može `pdflatex slika5` !!!
- ▶ put do pdf na više načina

gnuplot, parametric plot

- ▶ `set parametric`
- ▶ `plot [0:2*pi] cos(t), sin(t), 2*cos(t/2), 2*sin(t/2)`
- ▶ `set xrange [-3:3]`
- ▶ `set yrange [-3:3]`
- ▶ `replot`
- ▶ `set size square`
- ▶ `replot`
- ▶ `set key above box; replot`

... and a script file ...

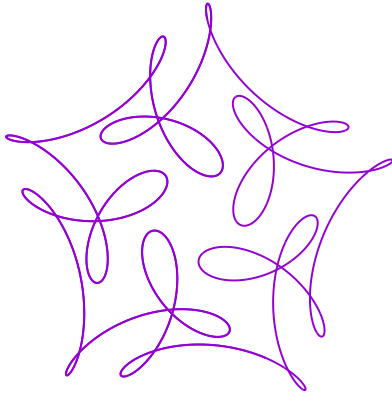
```
#!/bin/sh
gnuplot $1.gp
pdflatex $1.tex
rm -f $1.tex
rm -f $1-inc*
rm -f $1.aux
rm -f $1.log
```

save it as `gpmake.sh`, just run it,

```
sh gpmake.sh mystery-curve
```

and look for the output

... and the result is ...



gnuplot, polar plot

- ▶ reset
- ▶ set polar
- ▶ plot [0:2*pi] t
- ▶ plot [0:10*2*pi] t
- ▶ set samples 1000
- ▶ set size square
- ▶ set xrange [-70:70]
- ▶ set yrange [-70:70]
- ▶ unset key; replot
- ▶ plot [0:10*2*pi] t, -t

gnuplot, logscale, kompleksni brojevi

- ▶ resetujte gnuplot settings, restart to sigurno radi
- ▶ smatraćemo da je $x = \frac{\omega}{\omega_p}$
- ▶ j={0,1}
- ▶ print j**2
- ▶ print j*j
- ▶ set xrange [0.001:1000]
- ▶ set logscale x
- ▶ plot 20*log10(abs(1/(1+j*x)))
- ▶ set yrange [-80:20]; replot
- ▶ plot 20*log10(abs(1/(1+(j*x)**2)))
- ▶ set yrange [-160:160]; replot
- ▶ set ytics 20; replot
- ▶ set samples 1000; replot
- ▶ set samples 10000; replot

gnuplot, fazna karakteristika

- ▶ plot 180/pi*arg(1/(1+(j*x)**2))
- ▶ set yrange [-270:90]; replot
- ▶ set ytics 90; replot
- ▶ plot 180/pi*arg(1/(1+j*x))
- ▶ set yrange [-135:45]; replot
- ▶ set ytics 45; replot

gnuplot, amplitudska i fazna karakteristika, script

napravite skript af.gp:

```
set terminal wxt 0 persist
j={0,1}
set logscale x
set xrange [0.001:1000]
set yrange [-120:40]
set ytics 20 nomirror out
set y2range [-270:90]
set y2tics 90 nomirror out
set samples 1000
set xlabel "log(w/(1 rad/s))"
set ylabel "a [dB]"
set y2label "phi [deg]"
plot 20*log10(abs(1/(1+(j*x)**2))) axes x1y1, \
180/pi*arg(1/(1+(j*x)**2)) axes x1y2
```

gnuplot af.gp

gnuplot, styles and size

```
set xrange [-pi:pi]
set xtics pi
set yrange [-1.5:1.5]
set ytics 1
set style line 1 linetype 1 linewidth 4 linecolor rgb "red"
set style line 2 linetype 2 linewidth 2 linecolor rgb "blue"
plot sin(x) linestyle 1, cos(x) linestyle 2
set terminal postscript eps color size 12cm, 8cm
set output "size1.ps"
replot
set terminal postscript eps color size 4cm, 4cm
set output "size2.ps"
replot
```

gnuplot, margins

```
show margin
set terminal postscript eps color size 12cm, 8cm
set output "margin1.ps"
replot
set lmargin at screen 0.2
set rmargin at screen 0.9
set bmargin at screen 0.2
set tmargin at screen 0.9
show margin
set output "margin2.ps"
replot
set lmargin
show margin
```

gnuplot, axis, border, tics

```
reset
set terminal wxt 0
set output
plot sin(x)
unset border; replot
set zeroaxis; replot
set xtics axis; replot
set ytics axis; replot
set zeroaxis linetype -1; replot
```

gnuplot, test, terminal specific parameters

- ▶ vrlo korisna komanda, `test`
- ▶ prikazuje "terminal specific" parametre
- ▶ zaključak: odaberete nekoliko terminala koji vas interesuju (`epslatex`, `png`, `jpeg`, `postscript`, `pdfcairo`), naviknete se na značenje pojedinih parametara
- ▶ kroz `test` se odlično vidi razlika između vektorske i rasterske grafike

gnuplot, test, važniji terminali

```
set terminal png
set output "test.png"
test
set terminal postscript eps
set output "test.eps"
test
set terminal pdfcairo
set output "test.pdf"
test
set terminal jpeg
set output "test.jpg"
test
set terminal wxt 0
set output
test
```

gnuplot, size, ratio

```
reset
set terminal wxt 0
set output
set parametric
plot [0:2*pi] cos(t), sin(t)
set xrange [-2:2]
set yrange [-1:1]
replot
set size square
replot
set size ratio 1
replot
set size ratio 0.5
replot
```

gnuplot, wxt interactive commands

```
reset
plot exp(x)
l
l
m
m
```

gnuplot, 3D plot, sampling

```
reset
set xrange [-2*pi:2*pi]
set yrange [-2*pi:2*pi]
splot cos(x)*sin(y)
show isosamples
set isosamples 2; replot
show samples
set samples 10; replot
set samples 100; replot
set isosamples 3; replot
set isosamples 5; replot
set isosamples 7; replot
```

gnuplot, 3D plot, hidden3d

```
show hidden3d
set hidden3d
show hidden3d
replot
set isosamples 50
replot
```

gnuplot, 3D plot, contours

```
show contour
set contour
show contour
replot
help contour
set contour both
replot
```

gnuplot, 3D plot, surface

```
set contour surface
replot
unset surface
replot
```

gnuplot, 3D plot, more contours

```
splot sin(sqrt(x**2+y**2))/sqrt(x**2+y**2) title "sombbrero"  
help cntrparam  
set cntrparam levels discrete 0,0.25,0.5,0.75  
replot  
set surface  
replot  
set contour both  
replot
```

gnuplot, 3D plot, cubic spline

```
set isosamples 10  
replot  
set cntrparam cubicspline  
replot  
show view  
set view map  
replot  
unset surface  
replot  
set size ratio 1  
replot  
set isosamples 50  
replot
```

gnuplot, 3D plot, views

```
show view  
set view  
show view  
replot
```

gnuplot, 3D plot, bojenje

```
set pm3d  
replot  
unset contour  
replot  
set view map  
replot  
set view  
replot  
set pm3d at b  
replot  
set pm3d at t  
replot
```

gnuplot, 3D plot, 3D parametarski plot

```
reset  
set parametric  
splot [0:20*pi] cos(u), sin(u), u  
  
set samples 1000  
replot
```

octave, data files

```
octave  
  
z = rand(21);  
save z z  
x = linspace(0, 4 * pi, 361);  
x0 = cos(x);  
y0 = sin(x);  
z0 = exp(x / pi);  
xx = [x0' y0' z0'];  
save xx xx  
ctrl-d
```

gnuplot, data matrix plot

```
splot "z" matrix  
splot "z" matrix with lines  
set hidden3d; replot
```

gnuplot, contours

```
set contour; replot  
unset surface; replot  
set view map; replot  
set size ratio 1; replot  
unset key; replot  
unset tics; replot  
set cntrparam cubicspline; replot
```

gnuplot, palette

```
set pm3d; replot
unset colorbox; replot
unset contour; replot
set palette gray; replot
```

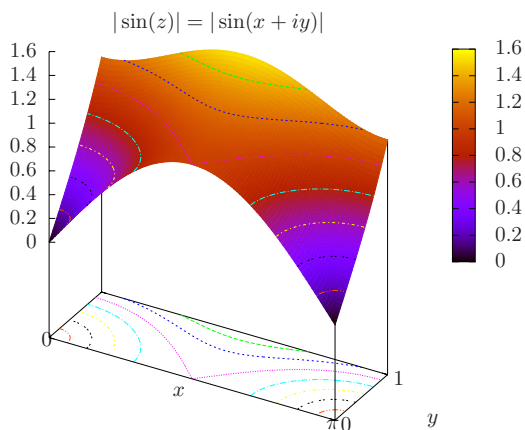
gnuplot, 3D plot, frekvencijski primer

```
j={0,1}
set xlabel "sigma"
set ylabel "j omega"
set xrange [-3:3]
set yrange [-3:3]
set zrange [0:5]
splot abs(1/(1+(x+j*y)**2))
```

gnuplot, frekvencijski primer, colorplot

```
set view; replot
set pm3d; replot
set pm3d at t; replot
set pm3d at st; replot
set pm3d at b; replot
set pm3d map; replot
```

gnuplot, set view equal, fig.gp, extended



gnuplot, 3D data line

```
reset
splot "xx" using 1:2:3
```

```
splot "xx" using 1:2:3 with lines title "spirala"
```

gnuplot, 3D plot, frekvencijski primer, contours

```
set contour; replot
set contour surface; replot
set contour both; replot
set cntrparam levels discrete 1,2,3; replot
unset surface; replot
set view map; replot
set xrange [-2:2]
set yrange [-2:2]
replot
set cntrparam cubicspline; replot
set size square; replot
```

gnuplot, set view equal

```
set xrange [0: pi]
set yrange [0:1]

set contour both
unset surface
set cntrparam levels discrete 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4
set pm3d

set isosamples 41

set xlabel "x"
set xtics ("0" 0, "pi" pi)
set ylabel "y"
set ytics ("0" 0, "1" 1)

set label at pi/2, 0.5, 2 "|sin(z)| = |sin(x + i y)|" center

set view equal

splot sqrt(sin(x)**2*cosh(y)**2+cos(x)**2*sinh(y)**2) notitle
```

gnuplot, planeta, 3D

```
set terminal wxt size 600,600 persist

unset key; unset border; unset tics

set lmargin screen 0.05; set bmargin screen 0.05
set rmargin screen 0.95; set tmargin screen 0.95

set mapping spherical
set angles degrees
set hidden3d
set xyplane at -1
set view 60, 120

set parametric
set isosamples 37
set urange[0:360]; set vrange[-90:90]
set size square

splot cos(v)*cos(u), cos(v)*sin(u), sin(v) w l lc rgb "cyan", \
"world_110m.txt" with lines lc rgb "blue" lw 2
```

gnuplot, planeta, 3D



gnuplot, planeta, 2D

```
set terminal pdfcairo size 360 * 0.01, 180 * 0.01
set output "map.pdf"

unset key
unset border
unset tics

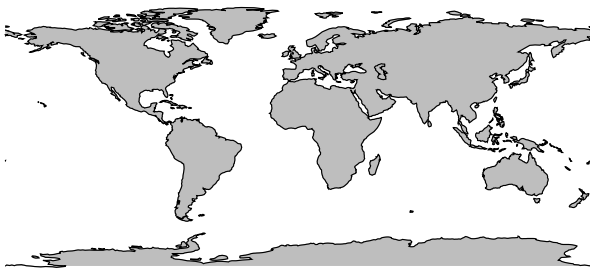
set style line 1 lc rgb "gray" lt 1 lw 1
set style line 2 lc rgb "black" lt 1 lw 1

set lmargin 0
set rmargin 1
set tmargin 1
set bmargin 0

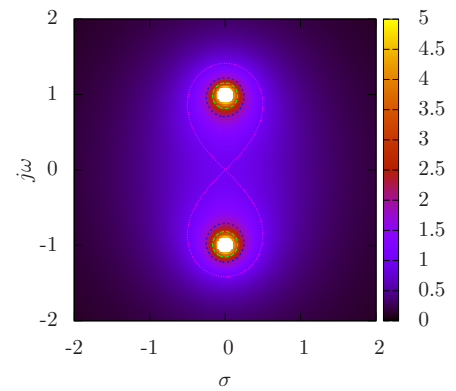
set xrange [-180:180]
set yrange [-90:90]

plot "world_110m.txt" w filledcu ls 1, \
     "world_110m.txt" w l ls 2
```

gnuplot, planeta, 2D



gnuplot, zaključak



preporuka: <http://gnuplot.sourceforge.net/demo/>

Maxima, uvod 1

Maxima, wxMaxima

- ▶ computer algebra systems (CAS)? http://en.wikipedia.org/wiki/Comparison_of_computer_algebra_systems
- ▶ **Mathematica**, \$2,495 (Professional), \$1095 (Education), \$140 (Student), \$69.95 (Student annual license), \$295 (Personal), **free on Raspberry Pi hardware**
- ▶ **Maple**, \$2,275 (Commercial), \$2,155 (Government), \$1245(Academic), \$239 (Personal Edition), \$99 (Student), \$79 (Student, 12-Month term)
- ▶ navijačke strasti, Maple vs. Mathematica
- ▶ Symbolic Math Toolbox (MATLAB), \$3150 including required Matlab
- ▶ bio popularan **Derive**, [http://en.wikipedia.org/wiki/Derive_\(computer_algebra_system\)](http://en.wikipedia.org/wiki/Derive_(computer_algebra_system)), discontinued 2007

Maxima, uvod 2

- ▶ Maxima, a computer algebra system, GPL
- ▶ [http://en.wikipedia.org/wiki/Maxima_\(software\)](http://en.wikipedia.org/wiki/Maxima_(software))
- ▶ <http://maxima.sourceforge.net/>
- ▶ zasnovana na MIT Macsyma, <http://en.wikipedia.org/wiki/Macsyma>
- ▶ William Frederick Schelter, GPL, DOE Macsyma 1982 http://en.wikipedia.org/wiki/Bill_Schelter
- ▶ star program, ali aktivno se razvija
- ▶ uključen u **Scilab** i **Euler Math Toolbox**
- ▶ uključen u **SageMath**, [http://en.wikipedia.org/wiki/Sage_\(mathematics_software\)](http://en.wikipedia.org/wiki/Sage_(mathematics_software))
- ▶ napredno: SageMath, <http://www.sagemath.org/>

Maxima, uvod 3

- ▶ Maxima je program komandne linije, startuje se sa maxima
- ▶ nekada ga pratio GUI xmaxima, još postoji (sudo apt-get install xmaxima)
- ▶ sticajem okolnosti, korišćićemo xmaxima i mi ove godine ...
- ▶ danas dominira GUI wxMaxima, Andrej Vodopivec
- ▶ <https://wxmaxima-developers.github.io/wxmaxima/>
- ▶ wxMaxima može sve što može Maxima, **nešto** (ne sve) uneto u GUI
- ▶ odlično radi i pod win
- ▶ treba predavati u **prvoj** godini (jeretička misao)
- ▶ **napomena**: problemi pod **nekim** verzijama GNU/Linux!
- ▶ zato ćemo izbegavati veliku primenu i koristiti i xmaxima

Maxima, literatura

- ▶ <http://maxima.sourceforge.net/documentation.html>
- ▶ Maxima Reference Manual, ver. 5.42.0, 1196 strana
- ▶ na istoj stranici niz linkova na tutorials
- ▶ lepi primeri <http://math.hawaii.edu/wxmaxima/>
- ▶ film od 2h 9' 9"(sedam delova), počinje sa <http://www.youtube.com/watch?v=iqVWZcxAsrs>

Maxima, komandna linija

- ▶ da počnemo sa komandnom linijom
- ▶ zašto?
- ▶ 1. programiranje moguće; nalik Octave, Python, interpreter
- ▶ 2. GUI nema sve komande, mnogo ih je
- ▶ 3. kada se naviknete, ponekad lakše je kucati komande
- ▶ **maxima**
- ▶ izlazi se sa **quit()**;

Maxima, komandna linija

- ▶ čudna i česta upotreba znaka %
- ▶ π je **%pi**
- ▶ e je **%e**
- ▶ j ili i je **%i**
- ▶ ∞ je **inf**, nema %, ima i **infinity** i **infm**
- ▶ **;** je kraj komande, nema veze sa Octave, čak suprotno!
- ▶ **\$** je kraj komande to supress output
- ▶ **float(<izraz>)**; daje numerički izlaz

Maxima, komandna linija ...

- ▶ **%pi**;
- ▶ **float(%pi)**;
- ▶ **ev(%pi, numer)**;
- ▶ **%e**;
- ▶ **float(%e)**;
- ▶ **float(%i)**;
- ▶ **%i^2**;
- ▶ **sqrt(-1)**;
- ▶ **sin(%pi/4)**;

simboli, brojevi, razlomci ...

- ▶ `3/4;`
- ▶ `3.0/4;`
- ▶ `3/4.;`
- ▶ `3/4.0;`
- ▶ `float(3/4);`
- ▶ `ev(3/4, numer);`
- ▶ `1/2 - 1/3;`
- ▶ `float(%);`
- ▶ `sin(2);`
- ▶ `sin(2.0);`
- ▶ `63^3;`
- ▶ `3^63;`
- ▶ `10!;`
- ▶ `100!;`
- ▶ `1000!;`

=, % i solve

- ▶ još jedna čudna primena %, prethodni izraz
- ▶ `%ok`, k -ti output, k je integer, $k \in N$
- ▶ `%ik`, k -ti input
- ▶ `%;`
- ▶ `%i2;`
- ▶ `%o2;`
- ▶ kod wxMaxima % može da vas prevvari, vrtićemo se na ovo, pazite!
- ▶ = je = iz jednačine, ne zaboravite, ponavljam!
 - ▶ `3*x+2=8;`
 - ▶ `solve(%, x);`
 - ▶ `solve(y^3 = 27, y);`
 - ▶ `solve(f(t)=64,t);`

kill

- ▶ `describe(kill);`
- ▶ `? kill;`
- ▶ `a: 4;`
- ▶ `b: 5;`
- ▶ `f(x):=x^2;`
- ▶ `values;`
- ▶ `kill(b);`
- ▶ `values;`
- ▶ `fundef(f);`
- ▶ `kill(f);`
- ▶ `fundef(f);`
- ▶ `b: 7;`
- ▶ `values;`
- ▶ `kill(all);`
- ▶ `values;`

fpprec i bfloat

- ▶ `fpprec;`
- ▶ `float(%pi);`
- ▶ `bfloat(%pi);`
- ▶ `fpprec: 50;`
- ▶ `bfloat(%pi);`
- ▶ `float(%pi);`
- ▶ `fpprec: 3;`
- ▶ `bfloat(%pi);`
- ▶ `float(%pi);`
- ▶ `describe(fpprec);`
- ▶ `describe(bfloat);`

problemi sa =, osnovna prepreka primeni

- ▶ = je = iz jednačine
 - ▶ `solve(2*x-8=2,x);`
 - ▶ `solve(x^2 - 5 * x = -6, x);`
 - ▶ `solve(x^2 - 2, x);`
- ▶ : je = iz dodele vrednosti
 - ▶ `a;`
 - ▶ `a: 4;`
 - ▶ `a;`
 - ▶ `a$`
 - ▶ `a^3;`
 - ▶ `sqrt(a);`
- ▶ := je = iz definicije funkcije
 - ▶ `f(x):=x^2;`
 - ▶ `f(3);`
 - ▶ `f(a);`
 - ▶ `f(b);`

undefinisanje

- ▶ `fundef(f);`
- ▶ `remfunction(f);`
- ▶ `fundef(f);`
- ▶ može i `remfunction(all);`
- ▶ `values;`
- ▶ `remvalue(a);`
- ▶ `a;`
- ▶ `a: 2;`
- ▶ `b: 3;`
- ▶ `values;`
- ▶ `remvalue(all);`
- ▶ `values;`
- ▶ ovde posle ? mora blanko:
- ▶ `? values;`
- ▶ `? remvalue;`

još o funkcijama i ev

- ▶ `f(x) := x^2;`
- ▶ `a: x^2;`
- ▶ `f(y);`
- ▶ `ev(a, x=y);`
- ▶ `f(4);`
- ▶ `ev(a, x=4);`
- ▶ `describe(ev);`
- ▶ opet neobična primena simbola =

fpprintprec

- ▶ `fpprintprec;`
- ▶ `float(%pi);`
- ▶ `float(%e);`
- ▶ `fpprintprec: 3;`
- ▶ `float(%pi);`
- ▶ `float(%e);`
- ▶ `fpprintprec: 5;`
- ▶ `float(%pi);`
- ▶ `float(%e);`
- ▶ `fpprintprec: 0;`
- ▶ `float(%pi);`
- ▶ `float(%e);`
- ▶ `describe(fpprintprec);`

expand i factor

- ▶ `expand((x+1)^2);`
- ▶ `expand((x+1)*(x-1));`
- ▶ `expand((x-3)^7);`
- ▶ `factor(%);`
- ▶ `eq: expand((x-4)*(x-5)*(x-6));`
- ▶ `solve(eq, x);`
- ▶ `factor(eq);`
- ▶ `factor(4*x^5-4*x^4-13*x^3+x^2-17*x+5);`
- ▶ `factor(1001);`
- ▶ `factor(123412341234);`
- ▶ `describe(factor);`
- ▶ `describe(expand);`

parcijalni razlomci, `partfrac(expression, variable);`

- ▶ `ex: (s^3+4*s^2+6*s+4)/(s^3+3*s^2+3*s+1);`
- ▶ `factor(ex);`
- ▶ `partfrac(ex, s);`
- ▶ dobar TEK!
- ▶ `expand(%);`
- ▶ `ratsimp(%);`

ratsimp i fullratsimp

- ▶ `kill(all);`
- ▶ `eq: sin(x/(x^2+x)) = exp((log(x)+1)^2-log(x)^2);`
- ▶ `ratsimp(eq);`
- ▶ `((x-1)^(3/2)-(x+1)*sqrt(x-1))/sqrt((x-1)*(x+1));`
- ▶ `ratsimp(%);`
- ▶ `expr: (x^(a/2)+1)^2*(x^(a/2)-1)^2/(x^a-1);`
- ▶ `ratsimp(%);`
- ▶ `fullratsimp(%);`
- ▶ `? ratsimp;`
- ▶ `? fullratsimp;`

trigonometrija

- ▶ `cos(%pi/3);`
- ▶ `sin(%pi/3);`
- ▶ `ev(sin(%pi/3), numer);`
- ▶ `float(sin(%pi/3));`
- ▶ `csc(45*%pi/180);`
- ▶ `tan(%pi/4);`
- ▶ `tan(%pi/8);`
- ▶ `acos(1/2);`
- ▶ `180/%pi*asin(sqrt(3)/2);`
- ▶ `acsc(1);`

trigonometrija, izrazi

- ▶ `ex: sin(x)^2+cos(x)^2;`
- ▶ `trigsimp(ex);`
- ▶ `? trigsimp;`
- ▶ `kill(all);`
- ▶ `ex: sin(a+b);`
- ▶ `trigexpand(ex);`
- ▶ `trigrat(%);`
- ▶ `? trigexpand;`
- ▶ `? trigrat;`
- ▶ `ex: sin(x)^2;`
- ▶ `trigsimp(ex);`
- ▶ `trigreduce(ex);`
- ▶ `? trigreduce;`
- ▶ `trigrat(ex);`

linearni sistemi jednačina 1

- ▶ `3*x+2*y=7;`
- ▶ `lhs(%);`
- ▶ `rhs(%);`
- ▶ `rhs(%ok);` *k je broj jednačine iz prvog reda*
- ▶ `e1: 3*x+2*y=7;`
- ▶ `lhs(e1);`
- ▶ `rhs(e1);`
- ▶ `e2: 5*x-y=3;`
- ▶ `linsolve([e1,e2], [x,y]);`
- ▶ `r: %;`
- ▶ `r[1];`
- ▶ `r[2];`
- ▶ `rhs(r[1]);`

linearni sistemi jednačina 2

- ▶ `e1;`
- ▶ `e1: 10*x-2*y=6;`
- ▶ `e2;`
- ▶ `linsolve([e1,e2], [x,y]);`
- ▶ `e1: 10*x-2*y=5;`
- ▶ `linsolve([e1,e2], [x,y]);`

eliminacija

- ▶ `remvalue(all);`
- ▶ `a1: x+y+2*t=7;`
- ▶ `a2: x-y-t=2;`
- ▶ `eliminate([a1,a2], [t]);`

matrice

- ▶ `A: matrix([1,2],[2,1]);`
- ▶ `B: invert(A);`
- ▶ `A*B;`
- ▶ ovo je element-wise operation!!!
- ▶ `A.B;`
- ▶ ovo je matricno množenje!
- ▶ `determinant(A);`
- ▶ `determinant(B);`
- ▶ `b: matrix([3],[3]);`
- ▶ `x: B.b;`
- ▶ `A.x;`
- ▶ `A.x-b;`
- ▶ `kill(all);`

Kronecker-Capelli

- ▶ `A: matrix([5,-1,3],[10,-2,6]);`
- ▶ `echelon(A);`
- ▶ `A[2][3]: 5;`
- ▶ `A;`
- ▶ `echelon(A);`

nelinearni sistemi

- ▶ `remvalue(all);`
- ▶ `a1: x^2+y^2=41;`
- ▶ `a2: y=x+1;`
- ▶ `algsys([a1,a2],[x,y]);`
- ▶ `t: solve(a1,y);`
- ▶ `y1: rhs(t[1]);`
- ▶ `y2: rhs(t[2]);`
- ▶ `solve(a2,y);`
- ▶ `y3: rhs(solve(a2,y)[1]);`

plotovanje

- ▶ pazite, wxMaxima ima i wxplot2d koji embed graphs!
- ▶ `plot2d([y1,y2,y3],[x,-10,10],[y,-15,15]);`
- ▶ stari poznanik, gnuplot
- ▶ Octave radi isto, gnuplot
- ▶ `describe(plot2d);`
- ▶ ima i `example(linsolve);`
- ▶ `%th(3);`
- ▶ probajte `%th(4);`

limesi

- ▶ `limit((1+1/x)^(2*x),x,inf);`
- ▶ `float(%);`
- ▶ `(x-2)/(x^2-4);`
- ▶ `limit(%,x,2);`
- ▶ `limit(x^3,x,inf);`
- ▶ `limit(x^3,x,minf);`
- ▶ `f(x) := atan(x);`
- ▶ `limit(f(x),x,inf);`
- ▶ `limit(f(x),x,minf);`
- ▶ `limit(sin(3*x)/x,x,0);`

0, 0- i 0+

- ▶ `limit(1/x,x,0);`
- ▶ `limit(1/x,x,0,plus);`
- ▶ `limit(1/x,x,0,minus);`

kombinacije ...

- ▶ `f(x):=x^3*tan(x);`
- ▶ `rd: (f(x+h)-f(x))/h;`
- ▶ `limit(rd,h,0);`
- ▶ `trigsimp(%);`
- ▶ `trigrat(%);`

izvodi

- ▶ `remvalue(all);`
- ▶ `diff(x^2,x);`
- ▶ `diff(sin(x),x);`
- ▶ `diff(sin(x),x,2);`
- ▶ `diff(sin(x),x,3);`
- ▶ `diff(sin(x),x,4);`
- ▶ parcijalni izvodi
- ▶ `diff(sin(x*y),x);`
- ▶ `diff(sin(w*t),t);`

razvoj u red

- ▶ `taylor(sin(x), x, 0, 5);`
- ▶ `taylor(cos(x), x, 0, 7);`
- ▶ `f(x) := %e^x-cos(x);`
- ▶ `taylor(f(x), x, 0, 7);`
- ▶ vezba: nacrtajte $\sin(x)$ i razvoje reda 1, 3, 5 i 7 na istom grafiku

operator '

- ▶ `kill(all);`
- ▶ `a: 4;`
- ▶ `a;`
- ▶ `'a;`
- ▶ ovo će biti potrebno kod diferencijalnih jednačina
- ▶ `'diff(x^2,x);`
- ▶ `'integrate(x^2,x);`
- ▶ `'integrate(x^2,x,0,1);`

provera rešenja, substitute

- ▶ `eq: 'diff(y, t, 2) + y = 0;`
- ▶ `s: ode2(eq, y, t);`
- ▶ `s: rhs(s);`
- ▶ `p: subst(s, y, eq);`
- ▶ `ev(p, diff);`
- ▶ `is(%);`

Laplasova transformacija

- ▶ `kill(all);`
- ▶ `laplace(1, t, s);`
- ▶ `laplace(sin(w*t), t, s);`
- ▶ `laplace(cos(w*t), t, s);`
- ▶ `laplace(exp(a*t), t, s);`
- ▶ `laplace(exp(a*t)*sin(w*t), t, s);`
- ▶ `laplace(exp(a*t)*cos(w*t), t, s);`
- ▶ `laplace(exp(t-T), t, s);`

integrali

- ▶ neodređeni integral
- ▶ `integrate(x^2, x);`
- ▶ `integrate(sin(x), x);`
- ▶ određeni integral
- ▶ `integrate(x^2, x, 1, 2);`
- ▶ `integrate(sin(x), x, 0, %pi);`
- ▶ `integrate(1/(1+x^2), x, 0, 1);`

uvod u diferencijalne jednačine

- ▶ `eq1: 'diff(y, t, 2) + 4 * y = 0;`
- ▶ `ode2(eq1, y, t);`
- ▶ `eq2: 'diff(y, t, 2) - 4 * y = 0;`
- ▶ `ode2(eq2, y, t);`
- ▶ `eq3: 'diff(y, t, 2) - 2 * 'diff(y, t) + y = 0;`
- ▶ `ode2(eq3, y, t);`
- ▶ `eq4: 'diff(y,t,2)+2*'diff(y,t)+4*y=8*sin(4*t);`
- ▶ `ode2(eq4, y, t);`

scripting

- ▶ pun programski jezik, input, output, if, for, ...
- ▶ samo naznake ovde, ako vas zanima dalje sami ...
- ▶ komandna linija, `gedit dj.mac`

```
eq1: 'diff(y, t, 2) + 4 * y = 0;
ode2(eq1, y, t);
eq2: 'diff(y, t, 2) - 4 * y = 0;
ode2(eq2, y, t);
eq3: 'diff(y, t, 2) - 2 * 'diff(y, t) + y = 0;
ode2(eq3, y, t);
eq4: 'diff(y, t, 2) + 2 * 'diff(y, t) + 4 * y
= 8 * sin(4*t);
ode2(eq4, y, t);
```

- ▶ Maxima prozor, `batch("dj.mac");`

inverzna Laplasova transformacija

- ▶ `ilt(1/(s+2), s, t);`
- ▶ `ilt(2/(s^2+4), s, t);`
- ▶ `ilt((s^2+3*s+3)/(s^3+3*s^2+3*s+1), s, t);`

wxMaxima

- ▶ realno, ovo se koristi, mada ima problema **ove** godine
- ▶ odličan GUI, ali jedan od, ima ih više, npr. xmaxima
- ▶ razlike na komandnoj liniji skoro da nema
- ▶ komandu šaljete sa ctrl/Enter ili sa desnim Enter
- ▶ postoji i wxplot2d i wxplot3d koji embed
- ▶ `plot3d(x^2-y^2, [x,-1,1], [y,-1,1]);`
- ▶ `wxplot3d(x^2-y^2, [x,-1,1], [y,-1,1]);`
- ▶ `a:2;`
- ▶ `b:3;`
- ▶ `c:4;`
- ▶ obrišete ćelije sa c i b
- ▶ `%;`
- ▶ pogledate okrećeni output, right click, Copy i Copy LaTeX

wxMaxima, zadaci 1

1. rešiti jednačinu $x^2 - 2x = 0$
2. rešiti jednačinu $ax^2 + 5bx - 2 = 0$
3. rešiti sistem jednačina $2x - y - 1 = 0$ i $x + 2y + 4 = 0$
4. nacrtati funkciju $y = e^x + 1$
5. imaju li preseke krug $x^2 + y^2 = 1$ i prave (algebra + grafici)
 - 5.1 $x + y - 4 = 0$
 - 5.2 $x + y - 1 = 0$
 - 5.3 $x + y - \sqrt{24} = 0$

wxMaxima, zadaci 3

1. Maklorenov red 2. i 4. stepena + slika za $f(x) = \ln(1 + x + x^2)$
2. Maklorenov red 6. stepena + slika za $f(x) = \ln(1 + \sin(x))$
3. Tejlorov red po $(x+3)$ 5. stepena za $f(x) = \sqrt{x^2 + 1}$
4. Tejlorov red po $(x-1)$ 5. stepena za $f(x) = \ln(x)$

wxMaxima, zadaci 5

- ▶ $\int \frac{\cos 2x}{\sin^2 x \cos^2 x} dx$
- ▶ $\int \left(\sin \frac{x}{2} - \cos \frac{x}{2} \right)^2 dx$
- ▶ $\int_1^e \frac{x^2 + 1}{2x} dx$
- ▶ $\int_3^9 x \sin x dx$
- ▶ $\int_0^\infty x e^{-x^2} dx$
- ▶ $\int_1^\infty \frac{2x}{(1+x^2)^2} dx$

wxMaxima

- ▶ pogledate Cell meni
- ▶ ćelije sa tekstem Ctrl+1 do Ctrl+4, valja komentarisati!
- ▶ pod win je ovo malo drugačije (Fsomething)
- ▶ pogledajte Save As ...
- ▶ pogledajte Export ...
- ▶ cilj#1: save mac file (manje važno, lako)
- ▶ cilj#2: print to pdf file
- ▶ cilj#3: export to LaTeX, posle pdf, ponekad bug!
- ▶ na ispitu/kolokvijumu output je pdf, iz wxMaxima!
- ▶ dalje vežbate sami, ja pomažem, lista zadataka ...

wxMaxima, zadaci 2

1. $\lim_{x \rightarrow \infty} \frac{x}{x^2 - 1}$
2. $\lim_{x \rightarrow 1} \frac{x}{x^2 - 1}$
3. naći prvi izvod $y = xe^x$
4. naći treći izvod $y = x^n e^x$
5. odrediti ekstreme i prevojne tačke funkcije $y = \frac{x^2}{x-2}$
6. nacrtati grafik funkcije $y = \frac{x^2}{x-2}$

wxMaxima, zadaci 4

- ▶ $\int \frac{3x+5}{x^2+x+1} dx$
- ▶ $\int xe^x dx$
- ▶ $\int_4^5 \frac{dx}{\sqrt[5]{x}}$
- ▶ $\int_0^{\pi/2} \frac{dx}{5+3\cos^2 x}$
- ▶ $\int_1^\infty \frac{1}{1+x^2} dx$
- ▶ $\int_1^\infty \frac{\ln^2 x}{x^2} dx$

wxMaxima, zadaci 6

- ▶ $\frac{dy}{dt} + y = \cos(t)$
- ▶ odrediti konstantu rešenja tako da $y(0) = \frac{1}{2}$
- ▶ napomene: koristiti subst i proveriti rešenje
- ▶ $\frac{dy}{dt} - 2y = -t^2$
- ▶ odrediti konstantu rešenja tako da $y(0) = \frac{1}{4}$
- ▶ $\frac{dy}{dt} + 2y = 2t$
- ▶ odrediti konstantu rešenja tako da $y(0) = -1$

Python?

Python, I deo

- ▶ programski jezik
- ▶ Wikipedia:
 - ▶ "Python is a general-purpose, high-level programming language whose design philosophy emphasizes code readability. Python claims to "[combine] remarkable power with very clear syntax", and its standard library is large and comprehensive. Its use of indentation for block delimiters is unique among popular programming languages."
 - ▶ "The reference implementation of Python (CPython) is free and open source software and has a community-based development model, as do all or nearly all of its alternative implementations. CPython is managed by the non-profit Python Software Foundation."

Python??

- ▶ interpreter, scripting language
- ▶ po tome nalik na BASIC (nekada), Octave, ...
- ▶ nema kompilacije i linkovanja, vrlo brze probe
- ▶ sporije od C-a
- ▶ ali se dobro povezuje sa C-om
- ▶ jako moćne i raznovrsne biblioteke (pySerial, numpy, matplotlib, sympy, ...)
- ▶ jednostavna sintaksa
- ▶ opšta namena
- ▶ free!!!
- ▶ jako dobro podržan, razvija se, rasprostranjen
- ▶ Google, Youtube, ...
- ▶ svaka distribucija GNU/Linux-a ga ima

Python???

- ▶ Guido van Rossum, December 1989
- ▶ masovno se uči kao prvi programski jezik: MIT, CU Boulder, ...
- ▶ radi pod raznovrsnim platformama, sve koje se kod nas sreću obuhvaćene
- ▶ vrlo objektno orijentisan, mada ne mora da se koristi
- ▶ vrlo moćni tipovi podataka
- ▶ lako se prave novi tipovi podataka

Python, kako nabaviti? GNU/Linux

GNU/Linux:

- ▶ već ima interpreter, sigurno
- ▶ provera: komandna linija, `python` ili `python3`
`Python 2.7.15rc1 (default, Nov 12 2018, 14:31:15)`
`[GCC 7.3.0] on linux2`
Type `"help"`, `"copyright"`, `"credits"` or `"license"` for more information.
- ▶ nešto valja dovući iz repository:
 - ▶ IDLE
 - ▶ IPython
 - ▶ numpy
 - ▶ scipy
 - ▶ matplotlib
 - ▶ pylab (sve prethodno)
 - ▶ python-serial
 - ▶ Sympy
 - ▶ Spyder
 - ▶ ...

Python, kako nabaviti? win

Windows:

- ▶ <http://python.org/>
- ▶ odaberete platformu, dovcete, instalirate
- ▶ za win je IDLE included
- ▶ ostalo?
 - ▶ <http://www.enthought.com/>
 - ▶ `ipython+numpy+scipy+matplotlib+...`
 - ▶ Canopy, zapravo PyLab
 - ▶ pySerial, SourceForge, <https://pypi.python.org/pypi/pyserial>
 - ▶ Sympy, <http://sympy.org/en/index.html>
 - ▶ Spyder, <https://pypi.python.org/pypi/spyder>
 - ▶ ...

Python, 2 ili 3?

- ▶ forking, 3 je „nov“ jezik
- ▶ 3 nema backward compatibility
- ▶ nisu prevelike razlike (print, za početak)
- ▶ problem sa već napisanim programima
- ▶ problem ako se oslanjate na već postojeće programe
- ▶ koristim numpy, matplotlib, ... pylab
- ▶ predajem verziju 2
- ▶ verziju 3 učite lako
- ▶ `python3`

Python, dokumentacija

- ▶ <http://www.python.org/doc/>, sve što treba
- ▶ <http://ocw.mit.edu/>, kurs 6.00 i 6.189
- ▶ isto, edX
- ▶ <http://openbookproject.net/thinkcs/python/english2e/>
- ▶ <http://greenteapress.com/thinkpython/thinkpython.html>
- ▶ još mnogo free resursa, realno je samo #1 potrebno
- ▶ izbor izlistan na sajtu predmeta

Python 3.6.7 (default, Oct 22 2018, 11:32:17)
[GCC 8.2.0] on linux
Type `"help"`, `"copyright"`, `"credits"` or `"license"` for more information.

Python, dokumentacija, realno

Ako ne učite programiranje, već programski jezik:

- ▶ <http://www.python.org/doc/>
- ▶ A4, pdf, zip, 11 MB
- ▶ Python 2.7.15, January 02, 2019
- ▶ tutorial.pdf, Python Tutorial, 149 strana
- ▶ reference.pdf, The Python Language Reference, 136 strana
- ▶ library.pdf, The Python Library Reference, 1584 strane
- ▶ ↑ ovde je suština uspeha

valja pomenuti i ...

- ▶ PyCharm
- ▶ <https://www.jetbrains.com/pycharm/>
- ▶ popularno ...
- ▶ Editions:
 1. Professional (proprietary!)
 2. Community ("Open Source")
- ▶ ne koristim ga, nekompetentan za komentare
- ▶ vidim da ga ljudi koje cenim vole i koriste ...
- ▶ ima tutorial i educational tools, PyCharm Edu

Python, počinjemo, kalkulator

Pokrenete IDLE ili ipython, kako god znate (kom. lin., dash, ...)

osnovne operacije:

```
2+2
2-3
2*3
```

a sada iznenađenje:

```
3/4*100
3.0/4.0*100.0
3.0/4*100
3./4*100
```

Python, da račistimo celobrojno deljenje

```
help(type)
type
type()
type(3)
type(3.0)
type(3.)
type(10/3)
type(10.0/3)
type(10/3.)
type(10./3.)
```

ovde se Python 3.x.x razlikuje!!!

Python, mislili ste da je sa deljenjem gotovo?

```
10.0/3.0
10.0//3.0
-10.0//3.0
-10/3
```

Python, stepenovanje i long

```
2^3
3^2
3^3
10^10
2**3
2 ** 3
3 ** 2
10 ** 10
type(10**10)
3**64
type(3**3)
type(3**64)
```

Python, ostatak pri celobrojnem deljenju

```
10%3
11%3
12%3
t=54+12
print t
type(t)
s=t/60
m=t%60
print s
print m
print s, m
print 'proteklo je', s, 'sat i', m, 'minuta'
```

Python, operatori poređenja

```
2 == 2
2==2
3 == 2
2 != 3
2 != 2
2 <> 2
2 <> 3
2 > 3
2 < 3
2 >= 1
2 >= 2
2 >= 3
2 <= 1
2 <= 2
2 <= 3
```


Python, logičke operacije, :, \ i

```
type(True); type(False)
a = True
b = False
type(a)
a and b # logicko i
not a # logicko ne
a and a
a or not a
a or (not a)
a or \
not b # ovako se nastavlja red
```

Python, zapisi brojeva

```
012
0o12
0012
0x35
0X35
0b11
0B11
```

Python, konverzija zapisa brojeva

```
oct(10)
hex(53)
bin(3)
```

Python, da račistimo ^, operacije nad bitima

```
a = 0b0101
a
b = 0b0011
b
a & b
bin(a & b)
bin(a | b)
bin(a ^ b)
bin(0)
bin(~0)
bin(2)
bin(~2)
~2
2 << 1
2 << 4
32 >> 2
3 >> 1
```

Python, a sada nesto sasvim drugačije: kompleksni brojevi

```
j*j
1j*1j
2J * 2J
type(1J)
abs(3+4j)
complex(1,2)
a = 2 + 3j
type(a)
a.real
a.imag
a.conjugate()
a * a.conjugate()
del a
type(a)
```

Python, malo ozbiljnija matematika, moduli

```
sin(1)
import math
type(math)
dir(math)
help(math)
help(math.sin)
math.sin(1)
math.e
math.pi
math.sin(math.pi/2)
math.exp(math.pi*1j)+1
math.cos(math.pi) + 1j * math.sin(math.pi) + 1
```

Python, namespaces

```
del math
import math as m
m.sin(m.pi / 4) ** 2
m.exp(1) - m.e
del m
from math import *
sin(pi / 4) ** 2
exp(1) - e
e
e = 32
e
pi
pi = 14
pi
```

Python, assignment operators

```
a = 1
a += 1
print a
a *= 2
print a
a /= 2
print a
a -= 4
print a
a **= 3
a %= 3
print a
-8 / 3
a = 11.0
a // 3
print a
```

Python, funkcije

```
def pdv(x):
    return x * 1.20

type(pdv)
pdv(100)
pdv(150)
```

Python, funkcije, help u više redova

```
def pdv(x):
    '''ovo je funkcija koja racuna pdv

    a pdv je porez na dodatu vrednost'''
    return x * 1.20

pdv(100)
help(pdv)
```

Python, kontrola toka

```
def parnost(n):
    if n/2*2 == n:
        print 'paran'
    else:
        print 'neparan'

parnost(4)
parnost(5)
parnost(4.2)
parnost(5.1)
```

Python, ispitivanje tipa, sada radi

```
def parnost(n):
    if str(type(n)) != "<type 'int'>":
        print 'argument nije ceo broj'
        return
    if n/2*2 == n:
        print 'paran'
    else:
        print 'neparan'

parnost(4.2)
parnost(4)
parnost(3)
```

Python, funkcije, help

```
def pdv(x):
    'ovo je funkcija koja racuna pdv'
    return x * 1.20

pdv(100)
help(pdv)
```

Python, funkcije, opcioni argumenti

```
def pdv(x, stopa = 20):
    return x*(1 + stopa/100)

pdv(100)
pdv(150)

def pdv(x, stopa = 20):
    return x * (1 + stopa/100.)

pdv(100)
pdv(150)
pdv(100, stopa=23)
pdv(100, 23)

del pdv
pdv(10)
```

Python, ispitivanje tipa

```
def parnost(n):
    if type(n) != "<type 'int'>":
        print 'argument nije ceo broj'
        return
    if n/2*2 == n:
        print 'paran'
    else:
        print 'neparan'

parnost(4.2)
parnost(4)
parnost(3)

type(4)
type(type(4))
type("<type 'int'>")
```

Python, ispitivanje tipa, može i ovako

```
def parnost(n):
    if type(n) != type(1):
        print 'argument nije ceo broj'
        return
    if n/2*2 == n:
        print 'paran'
    else:
        print 'neparan'

parnost(4.2)
parnost(4)
parnost(3)
parnost(4.)
```

Python, konverzije tipova i još ponešto

```
del parnost
int(-4.2)
int(4.2)
long(_)
float(_)
float(5)
divmod(10, 3)
divmod(12, 3)
pow(2, 8)
2 ** 8
str(float(2**8))
```

Python, liste

```
a = [1, 2, 5, 6]
type(a)
a[0]
a[1]
a[2]
a[3]
a[4]
a[-1]
a[-2]
a[-3]
a[-4]
a[-5]
print a
len(a)
```

Python, liste, slicing and mutability

```
a[1:3]
a[1 : 2]
a[1 : - 2]
a[2 : ]
a[:2]
a[:-2]
a[3] = 7
print a
```

Python, liste, dodavanje i brisanje elemenata

```
a + 9
a + [9]
a = a + [9]
len(a)
del a[(len(a) - 1)]
print a
del a[1]
print a
len(a)
```

Python, liste, metodi append i extend

```
a = [1, 2, 3, 4]
a.append(5)
print a
b = [6, 7]
a.append(b)
print a
len(a)
del a[5]
a.extend(b)
print a
len(a)
del a[5:]
print a
```

Python, liste, range

```
a = range(5)
len(a)
print a
a = range(4, 10)
len(a)
print a
a = range(3, 10, 2)
print a
a = range(10, 0, -2)
print a
```

Python, stack

```
a = []
type(a)
a.append(1)
a.append(2)
a.append(3)
a.pop()
a.pop()
print a
a = range(10)
a.pop(3)
print a
```

Python, liste, insert

```
a = range(10)
a.insert(3, 4)
print a
a.insert(0, 1)
print a
a.insert(len(a), 'kraj')
print a
```

Python, liste, reverse, sort

```
a = range(10)
a.reverse()
print a
a.reverse()
print a
a = [3, 4, 2, 1]
a.sort()
print a
```

Python, in operator

```
3 in a
4 in a
a.remove(4)
4 in a
```

Python, aliases

```
a = 3
b = a
a is b
a == b
id(a)
id(b)
help(id)
b += 1
a == b
a is b
id(a)
id(b)
```

Python, matrice

```
a = [[1, 2], [3, 4]]
len(a)
len(a[1])
print a[1][1]
print a[0][0]
a[0, 0]
```

Python, liste, brojanje i brisanje

```
a = [3, 2, 3, 1, 4, 3, 2, 2, 5, 2]
a.count(2)
a.count(3)
a.remove(3)
a.count(3)
print a
a.remove(3)
print a
a.remove(3)
print a
a.remove(3)
```

Python, liste, index metod

```
print a
a.index(2)
a.index(5)
a.index(1)
a.index(3)
```

Python, aliases with lists

```
a = [1, 2, 3]
b = a
a is b
a == b
b[1] = 0
a == b
print a
a is b
c = a[:]
c == a
c is a
c[1] = 2
c == a
print c
print a
```

Python, inicijalizacija nizova

```
a = []
print a
a = [0] * 10
print a
a = [[1] * 3] * 3
print a
```

Python, for petlja

```
a = range(10)
for i in a:
    print i + 1, '/', len(a)
```

Python, for petlja, over string, "iterable"

```
a = 'neobicno bas'
for znak in a:
    print znak
```

Python, if-else

```
a = 'abrakadabra'
b = ''
for znak in a:
    if znak != 'a':
        b += znak
    else:
        b += '_'
print b
```

Python, if-elif-else

```
a = 'abrakadabra'
b = ''
for znak in a:
    if znak == 'a':
        b += '_'
    elif znak == 'k':
        b += '*'
    else:
        b += znak
print b
```

Python, II deo

How to Think Like a Computer Scientist
Learning with Python

Allen Downey, Jeffrey Elkner, Chris Meyers

Green Tea Press, Wellesley, Massachusetts

... i jedan citat:

strana 90:

... One of the characteristics of algorithms is that they do not require any intelligence to carry out. They are mechanical processes in which each step follows from the last according to a simple set of rules.

In our opinion, it is embarrassing that humans spend so much time in school learning to execute algorithms that, quite literally, require no intelligence.

On the other hand, the process of designing algorithms is interesting, intellectually challenging, and a central part of what we call programming.

Python, dosta kalkulatora: programiranje!

- ▶ pokrenete IDLE
- ▶ File, New File (Ctrl + N)
- ▶ kucate program
- ▶ sintaksna provera: Run, Check Module (Alt + X)
- ▶ start programa: Run Module (F5)

Python, for petlja, continue

```
a = []
for i in range(100):
    if i % 4 != 0:
        a.append(i)
        continue
    print 'izbacen', i
print
print a
```

Python, for petlja, da dodamo i break

```
a = []
for i in range(100):
    if i % 4 != 0:
        a.append(i)
        continue
    print 'izbacen', i
    if (i+1) % 77 == 0:
        break
print
print a
```

Python, input

```
x = input('unesi neki podatak: ')
print type(x)
print x
```

probati: 1, 3.0, 'podatak'

Python, raw_input

```
x = raw_input('unesi neki podatak: ')
print type(x)
print x
```

probati: 1, 3.0, 'podatak'

Python, formatted print

calculator mode, again

```
print '{}'  
print '{}'  
print '{}'.format('Pera')  
print '{}, {}'.format('Mika', 'Laza')  
print '{0}, {1}'.format('Mika', 'Laza')  
print '{1}, {0}'.format('Mika', 'Laza')  
print '{1}, {1}'.format('Mika', 'Laza')
```

Python, formatted print, stari sistem

```
print 'pi = %5.3f' % math.pi  
print 'pi = %7.4f' % math.pi  
n = 9  
print 'polagalo je %d studenata' % n  
print 'polagalo je %10d studenata' % n
```

vuče se po starim programima
izbegavati!

Python, nije sve u matematici (?)

```
import string as s  
dir(s)  
help(s)  
s.ascii_letters  
s.printable  
s.uppercase  
s.digits  
s.octdigits
```

Python, OŠ, 1

Vreme je za zadatak:

- ▶ ulazni podatak je broj, $n < 20$ (npr.)
- ▶ treba odštampati LEPU (formatiranu) tablicu množenja

Polako, sada imamo preča posla ...

Python, formatted print, brojevi

```
print 'p = {}, q = {}'.format(3, 4)  
print 'p = {}, q = {}'.format(3.0, 4)  
print 'p = {1}, q = {0}'.format(3, 4)  
import math  
print 'pi = {:.2f}'.format(math.pi)  
print 'pi = {:.5f}'.format(math.pi)  
print 'pi = {:.10.5f}'.format(math.pi)  
print 'n = {:10d}'.format(100)  
print 'n = {:4d}'.format(100)  
print 'n = {:3d}'.format(100)  
print 'n = {:1d}'.format(100)
```

ima još mogućnosti, pogledajte Tutorial
postoji i formatiranje sa %, staro (next slide)
koristiti str.format() metod za nove programe!

Python, stringovi, još ponešto

```
a = 'Petar'  
b = 'Marko'  
len(a); len(b)  
a + b  
a * 3  
3 * a  
(a + 2 * b) * 3  
a - b  
a == a  
a == b  
a < b  
b < a  
'1' < '2'  
'1' < 2
```

Python, string methods

```
del s  
  
'Pera'.upper()  
'Pera'.lower()  
'Pera'.center(20)  
'Pera'.isalpha()  
'Pera6'.isalpha()  
  
'kako da ne'.capitalize()  
'kako da ne'.title()  
'kako da ne'.split()
```

i tako dalje, manual for string methods, pogledajte pre nego što
reinvent

Python, files, pisanje

```
f = open('proba.txt', 'w')  
print f  
type(f)  
f  
f.write('prvi red\n')  
f.write('drugi red\n')  
f.write('treći red\n')  
f.close # metod, obavezno()  
f.close()  
type(f)  
f
```

pogledajte fajl proba.txt

Python, files, čitanje

```
f = open('proba.txt', 'r')
print f
f
type(f)
a = f.read()
a
print a
f.close()
f
```

Python, files, čitanje po 10 bajtova

```
f = open('proba.txt', 'r')
f.read(10)
f.read(10)
f.read(10)
f.read(10)
f.close()
```

Python, files, čitanje po redovima

```
f = open('proba.txt', 'r')
f.readline()
f.readline()
f.readline()
f.readline()
f.close()
```

Python, OŠ, 2

Vreme je za zadatak:

- ▶ ulazni podatak je broj, $n < 20$ (npr.)
- ▶ treba formirati LEPU (formatiranu) tablicu množenja
- ▶ i zapisati je u .txt file

Polako, još uvek imamo preča posla ...

Python, OOP

OOP, Python's middle name

- ▶ korisnik definiše svoje tipove podataka, klase
- ▶ nad definisanim podacima definiše funkcije, tzv. metode, već ih videli na delu
- ▶ čisto i pregledno
- ▶ može, a ne mora
- ▶ šteta propustiti

Python, OOP, konvencije za dalje ...

- ▶ pokrenete IDLE, tu je **interpreter**
- ▶ otvorite **editor**: File, New Window, ...
- ▶ ono što kucate u interpreteru je **crveno**
- ▶ ono što kucate u editoru je **plavo**
- ▶ kucate "as is", kreativnost za sada nije na ceni
- ▶ file neka se zove `vector.py`

Python, class

kucate u editoru, kada završite proverite sintaksu i izvršite, to su Alt+X i F5

```
class Vector:
    pass
```

Python, class

```
type(Vector)
a = Vector()
type(a)
print a
a.x = 1
a.y = 2.0
a.z = 3j
type(a.x)
type(a.y)
type(a.z)
print a.x, a.y, a.z
del a
type(a.x)
```


Python, class help

```
help(Vector)
```

```
class Vector:
    """Vector(x, y, z)

    klasa koja definise vektore
    i operacije nad njima"""
```

note: nema više `pass`!

```
help(Vector)
```

Python, inicijalizacija, dodajemo na class, help string ostaje

inicijalizacija i se dodaje u class file, `vector.py` pazite na indent, inicijalizacija i class isto! 4 argumenta?

```
type(i)
print i
print i.x, i.y, i.z
```

```
x = Vector()
```

Python, inicijalizacija, dodajemo na class, help string ostaje

```
class Vector:
    . . .
    def __init__(self, x, y, z):
        self.x = x
        self.y = y
        self.z = z
```

```
i = Vector(1, 0, 0)
```

Python, inicijalizacija, default

```
class Vector:
    . . .
    def __init__(self, x = 0, y = 0, z = 0):
        self.x = x
        self.y = y
        self.z = z
```

```
i = Vector(1, 0, 0)
j = Vector(0, 1, 0)
k = Vector(0, 0, 1)
```

```
x = Vector()
print x.x, x.y, x.z
print i.x, i.y, i.z
help(Vector)
```

Python, string method, lepo prikazivanje instance

dodajete samo funkciju:

```
class Vector:
    . . .
    def __str__(self):
        return 'Vector(' + str(self.x) + ', ' + str(self.y) + ', ' + str(self.z) + ')'
```

```
help(Vector)
print i
print j
a = Vector(2, 2, 0)
print a
```

Python, operator overloading, sabiranje vektora

```
class Vector:
    . . .
    def __add__(self, other):
        return Vector(self.x + other.x, self.y + other.y, self.z + other.z)
```

```
help(Vector)
print i + j
print i + j + k

print Vector.__add__(i, j)
print i.__add__(j)
```

Python, operator overloading, oduzimanje vektora

```
class Vector:
    . . .
    def __sub__(self, other):
        return Vector(self.x - other.x, self.y - other.y, self.z - other.z)
```

```
help(Vector)
print i - j
print i - j - k

print Vector.__sub__(k, j)
print k.__sub__(j)
```

Python, da zakomplikujemo malo . . .

u interpreteru:

```
def sv(a, b):
    return a + b
```

```
print sv(i, j)
print sv(3, 4)
sv
type(sv)
```

```
del sv
```

```
sv = Vector.__add__
sv
type(sv)
```

```
print sv(i, j)
print sv(3, 4)
```

Python, opet operator overloading, negativni vektor

```
class Vector:
    . . .
    def __neg__(self):
        return Vector(-self.x, -self.y, -self.z)

help(Vector)
print -i
print -(-i)

print Vector.__neg__(k)
print k.__neg__()
```

Python, štampanje kao metod

```
class Vector:
    . . .
    def pr(self):
        print self

help(Vector)
help(Vector.pr)
i.pr()
i.pr()
Vector.pr(i)
k.__neg__().pr()
```

Python, apsolutna vrednost

```
class Vector:
    . . .
    def absval(self):
        return math.sqrt(self.x ** 2 + \
            self.y ** 2 + self.z ** 2)

help(Vector)
help(Vector.absval)
i.absval()
i.absval()
import math
i.absval()
Vector.absval(i)
a = Vector(2, 2, 0)
a.absval(); Vector.absval(a); a.absval() ** 2
a.x
```

Python, vektorski proizvod

```
class Vector:
    . . .
    def vecprod(self, other):
        x = self.y * other.z - self.z * other.y
        y = self.x * other.z - self.z * other.x
        z = self.x * other.y - self.y * other.x
        return Vector(x, y, z)

print Vector.vecprod(i, j)
print Vector.vecprod(j, i)
print i.vecprod(j)
print j.vecprod(i)
i.vecprod(j).pr()
j.vecprod(i).pr()
```

Python, skalarni (dot) proizvod

```
class Vector:
    . . .
    def dotprod(a, b):
        dp = a.x * b.x + a.y * b.y + a.z * b.z
        return dp

print Vector.dotprod(i, j)
print Vector.dotprod(j, i)
a = Vector(2, 2, 0)
print a.dotprod(j)
print a.dotprod(k)
a.dotprod(k).pr()
k.dotprod(i).pr() # zašto ne radi?
```

Python, proizvod množenja skalarom

```
class Vector:
    . . .
    def scalprod(self, other):
        return Vector(self.x * other, \
            self.y * other, self.z * other)

print Vector.scalprod(i, 3)
print Vector.scalprod(3, i)
print k.scalprod(3)
print 3.scalprod(k)
j.scalprod(3).pr()
3.scalprod(i).pr()
a = 3
a.scalprod(i).pr()
```

Python, overloaded *

```
class Vector:
    . . .
    def __mul__(self, other):
        return Vector.dotprod(self, other)

print i * 3
print i * i
print i * j
print i * k
print i.__mul__(i)
(i * i).pr()
i * i
i * j
```

Python, overoverloaded *

```
class Vector:
    . . .
    def __rmul__(self, other):
        return Vector.scalprod(self, other)

print 3 * i
print i * 3
print i * j
print i * i
print i.__rmul__(3)
i.__rmul__(3).pr()
```

Python, a sada poređenje, 1

```
class Vector:
    . . .
    def __cmp__(self, other):
        a = self.absval()
        b = other.absval()
        if a > b:
            return 1
        elif a < b:
            return -1
        else:
            return 0
```

Python, a sada poređenje, 2

```
a = Vector(2, 2, 2)
a > i
a == i
a < i
i > i
i == i
i != i
i < i
i == j
i == k
```

I ovo je kraj priče o vektorima, nema više red i blue!

Python, class and object variables

```
class Brojac:
    broj = 0
    def __init__(self):
        Brojac.broj += 1
        print 'sada ih ima', Brojac.broj
    def __del__(self):
        Brojac.broj -= 1
        print 'sada ih ima', Brojac.broj
```

Python, class and object variables, primena

```
a = Brojac()
b = Brojac()
c = Brojac()
d = Brojac()
del c
a = Brojac() # pise, pa brise
del a
del d
del b
```

Python, a sada nešto sasvim različito, map

```
x = range(11)
print x
def sq(a):
    return a ** 2

y = map(sq, x)
print y
```

Python, a sada nešto neočekivano

```
help([])
help(())
help({})
a = (1, 2, 3, 4, 5)
a[0]
a[3]
a[2] = 5
```

Python, dictionary

```
t = {'peja': 313, 'laza': 312, 'dule': 311}
print t
print t['peja']
t['peja'] = 314
print t
del t['peja']
print t
'dule' in t
'peja' in t
t.keys()
t['peja'] = 313
print t
```

Python, sets

```
a = 'abracadabra'
b = 'alacazam'
x = set(a)
print x
y = set(b)
print y
x - y
y - x
x | y
x & y
x ^ y
```

Python, exceptions, 0

```
def reciproc(x):
    try:
        return 1 / float(x)
    except:
        print 'Houston, we have a problem!'
```

Python, exceptions, 2

```
def reciproc(x):
    try:
        return 1 / float(x)
    except ZeroDivisionError:
        print 'delis nulom!'
    except ValueError:
        print 'nije dobar broj'

reciproc(3)
reciproc(3.)
reciproc(0)
reciproc(0.)
reciproc('a ovo?')
reciproc(1+0j)
```

Python, exceptions, 4

- ▶ "built-in" exceptions?
- ▶ <http://docs.python.org/2/library/exceptions.html>
- ▶ ovo je samo početak ...
- ▶ možete sami da definišete exception ...
- ▶ ili da raise exception ...
- ▶ ako vas zanima
<http://docs.python.org/2/tutorial/errors.html>

Python, encoding, 1

```
#!/usr/bin/python

print 'ja se ne zovem Čedomir Š. Petrović'
print 'ja se зовем Предраг Пејовић'
```

Python, exceptions, 1

```
def reciproc(x):
    try:
        return 1 / x
    except ZeroDivisionError:
        print 'delis nulom!'

reciproc(3)
reciproc(3.)
reciproc(0)
reciproc(0.)
reciproc('a ovo?')
```

Python, exceptions, 3

```
def reciproc(x):
    try:
        return 1 / float(x)
    except ZeroDivisionError:
        print 'delis nulom!'
    except (ValueError, TypeError):
        print 'nije dobar broj'

reciproc(3)
reciproc(3.)
reciproc(0)
reciproc(0.)
reciproc('a ovo?')
reciproc(1+0j)
```

Python, start programa

```
python ime_programa.py
```

ili

```
./ime_programa.py
```

akko je PRVA linija programa:

```
#!/usr/bin/python
```

i ako je urađeno

```
chmod +x ime_programa.py
```

čega se sećate, naravno; da izbegnete ./ dopunite PATH

Python, encoding, 2, mora drugi red!

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

print 'ja se ne zovem Čedomir Š. Petrović'
print 'ja se зовем Предраг Пејовић'
```

```
print '\n\n'  
print r'\n\n'  
print '\\n\\n\\n'
```

```
from __future__ import division  
  
i  
  
from __future__ import unicode_literals
```

Python, nije kraj, ovo je početak!

- ▶ ima još mnogo toga, ovo je samo osnova
- ▶ pre svega biblioteke, moduli . . .
- ▶ mnogo toga je gotovo, vaša je primena!
- ▶ razlika 2 i 3
- ▶ od sledeće godine 3?
- ▶ ostaju numpy, matplotlib, ipython, pylab, sympy
- ▶ a sada: tablica množenja

Šta je PyLab?

Python, III deo: PyLab i SymPy

- ▶ environment koji cine
 - ▶ NumPy
 - ▶ SciPy
 - ▶ Matplotlib
 - ▶ IPython
- ▶ kako se instalira pylab?
- ▶ Ubuntu: Software Center ili Synaptic
- ▶ win: <http://www.enthought.com/>
- ▶ ... Canopy

IDE? Spyder!

- ▶ potrebno?
- ▶ kako kome, kako za šta ...
- ▶ IPython meni sasvim dobar ...
- ▶ ... ima i Qt verziju, IPython Qt console
- ▶ ... mada postoji i Spyder
- ▶ <https://pypi.python.org/pypi/spyder>
- ▶ ako nema dovucite iz repository ...
- ▶ ima i pod win, ali ...

NumPy

- ▶ <http://www.numpy.org/>
- ▶ osnovna biblioteka za numerički zahtevne Python primene, sadrži:
 1. N-dimensional array object
 2. array slicing methods
 3. array reshaping methods
- ▶ i module za:
 1. basic linear algebra functions
 2. basic Fourier transforms
 3. advanced random number capabilities
- ▶ pokrenete IDLE
- ▶ `import numpy`
- ▶ `dir(numpy)`
- ▶ `help(numpy)`
- ▶ `del numpy`

NumPy

- ▶ obradićemo, sve, naravno, imamo vremena, ...
- ▶ evolutivno nastao, „haotično“ iz Numeric i NumArray
- ▶ znanje koje raste
- ▶ Travis E. Oliphant & Enthought, biće još kontakata sa njima
- ▶ <http://www.enthought.com/>
- ▶ nema načina da se zapamti i nije „zauvek“
- ▶ potreban nov način učenja i snalaženja
- ▶ izbeći reinventing
- ▶ masovno korišćenje help-a i primera
- ▶ [Matlab-Python-R](#)

SciPy

- ▶ <http://www.scipy.org/>
- ▶ scientific lib za Python, zavisi od NumPy
- ▶ nekoliko modula in a single package, kao i NumPy
- ▶ moduli za:
 1. statistics
 2. optimization
 3. numerical integration
 4. linear algebra
 5. Fourier transforms
 6. signal processing
 7. image processing
 8. ODE solvers
 9. special functions

SciPy

- ▶ i dalje u IDLE
- ▶ `import scipy`
- ▶ `dir(scipy)`
- ▶ `help(scipy)`
- ▶ `del scipy`
- ▶ dobro razmislite pre nego što krenete u reinvent!
- ▶ ja ovo ne mogu da zapamtim, sto zapamtim zaboravim
- ▶ pomenuh li nov način učenja?
- ▶ kako organizovati informacije, previše ih je (i kratko traju)
- ▶ “Numerical Recipes”

matplotlib

- ▶ <http://matplotlib.sourceforge.net/>, sjajan sajt
- ▶ package sa dugim nizom modula
- ▶ jako dobro se vidi strukturiran namespace, package.modul
- ▶ Python 2D plotting library (samo 2D?)
- ▶ ono što ima sjajno je, a sada ima skoro sve
- ▶ galerija i primeri
- ▶ uputstvo, 3.0.0, pdf, 2288 strana, 21.09.2018.
- ▶ John D. Hunter, video lectures
- ▶ sintaksa vrlo liči na Matlab i Octave
- ▶ skoro kao gnuplot, ali direktno, bez izlaženja u gnuplot

IPython

- ▶ <http://ipython.org/>
- ▶ <https://jupyter.org/>
- ▶ interaktivni Python environment, vrlo nalik na Octave, wxMaxima, ...
- ▶ autocompletion by tab
- ▶ doteruje komande da budu shvaćene
- ▶ od mnogo mogućnosti: store, history, logging, ...

PyLab

- ▶ sve prethodno zgodno spakovano
- ▶ sređen namespace da ne mora puno dot notation
- ▶ sintaksa jako liči na Octave/Matlab
- ▶ počinjemo, komandna linija, terminal
- ▶ `ipython --pylab`

PyLab, IPython, osnovno

```
help()
?
%quickref
a = 3
b = 'string'
lista = [1, 2, 3]
li<tab>
whos
```

PyLab, IPython, store

```
store a
store b
store lista
del a
del b
a
b
store -r
a
b
store -z
store
```

PyLab, matematika, konstante

```
pi
e
j
1j
exp(1j * pi)
math.exp(1j * pi)
help(exp)
help(math.exp)
exp?
math.exp?
e**(1j * pi) + 1
```

PyLab, matrice 1

```
a = array([[1, 2], [3, 4]])
a
a.size
a.shape
a.ndim
a.dtype
a.dtype.name
a.itemsize
a.transpose()
transpose(a)
a.T
```

PyLab, matrice 2

```
det(a)
eig(a)
b = eig(a)
type(b)
len(b)
type(b[0])
type(b[1])
c, d = b
c.size
c.shape
c.ndim
c.dtype.name
c.itemsize
```

PyLab, matrice 3, inverzija i množenje

```
aa = inv(a)
aa
aa.dtype.name
aa * a
a * aa
dot(a, aa)
dot(aa, a)
x = arange(10)
x
print x
dot(x, x)
```

Sve operacije su **elementwise!!!**
Velika razlika u odnosu na Octave!!!

Pylab, gde je dot, tu je cross

```
i = array([1, 0, 0])
j = array([0, 1, 0])
print dot(i, j), dot(j, i)
print cross(i, j)
print cross(j, i)
cross?
source(cross)
```

Poznato od nekud? Rekoh da već ima, ...
Teško je naći potpuno nov primer ...
Reinventing problem! Google pomaže puno!

Pylab, rang!!!

```
a = array([[1, 1], [1, 1]])
rank(a) # ???
help(rank) # sad jasno?
linalg.matrix_rank(a)
```

Pazite!!!
Nažalost, lično iskustvo, ne tuđe!

Pylab, inicijalizacija, neke posebne matrice

```
I = eye(3)
print I
nule = zeros(4)
print nule
zeros?
zeros((4, 2))
zeros(4, 2) # pazite!
zeros((4, 2), dtype = int)
zeros((4, 2), dtype = complex)
ones([3, 4])
empty([6, 4])
empty?
```

Pylab, još elementwise operacija

```
a + 1
a * a
a**3
a / 2
a / 2.
```

Pazite na /, velika razlika u odnosu na Octave!!!

Pylab, reshape

```
help(reshape)
print a
a.reshape(1, 4)
a.reshape(4, 1)
reshape(a, (1, 4))
reshape(a, (4, 1))
a.reshape(2, 3)
a.reshape(4)
a.reshape((4, ))
reshape(a, 4)
reshape(a, (4, ))
```

Pylab, in place array operations

```
xx = arange(20)
print xx
xx += 1
print xx
xx = xx.reshape(4, 5)
print xx
xx -= 1
print xx
xx *= 2
print xx
```

Pylab, transpose, revisited

```
help(transpose)
xx = arange(5) + 1
print xx
xx.shape
xx.ndim
xx = xx.transpose()
xx.shape
print xx
xx = xx.reshape(1, 5)
xx.ndim
print xx
xx.shape
xx = xx.transpose()
xx.shape
print xx
```

Pylab, eig, revisited

```
a = eye(2)
print a
b = eig(a)
type(b)
len(b)
c, d = b
print c
print d
a[0, 1] = 1
print a
b = eig(a)
c, d = b
print c
print d
```


Pylab, plot 1

```
x = arange(10) + 0.5 # nikako 1/2
y = x**2
plot(x, y)
plot(x, y, 'r+')
xlabel('x')
ylabel('y')
title('y = x**2')
```

pretty matlabeće, ali nema hold on
save kako hocete, odaberete format
u IDLE za ovo je bilo potrebno `ion()` (zaglavljivanje!)
pogledati `show()` i `ion()` help

```
close() ili close('all')
```

Pylab, plot 2

```
close('all')
phi = linspace(0, 2 * pi, 1000)
x = 2 * cos(phi)
y = 2 * sin(phi)
plot(x, y)
axis([-3, 3, -3, 3])
axis('equal')
axis([-3, 3, -3, 3])
axis([-4, 4, -3, 3])
title('kruznica')

axis?
```

Pylab, plot 3

```
close('all')

help(figure)

figure(figsize = (6, 6))
plot(x, y)
axis('equal')
axis([-3, 3, -3, 3])
title('kruznica')
```

Pazite na redosled 'equal' i [-3, 3, -3, 3]!!!

Pylab, plot 4, histograms

```
x = rand(10000)
plot(x)
close()
rand?

hist(x, 100)
close()

hist?
```

Pylab, plot 5, normal distribution

```
x = randn(10000)
plot(x)
close()

randn?

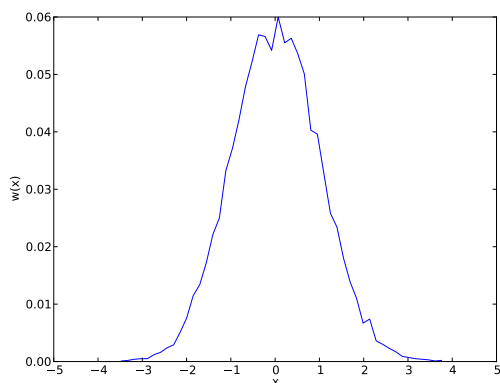
hist(x, 100)
close()

t = hist(x, 50)
type(t)
len(t)
len(t[0])
len(t[1])
```

Pylab, plot 6

```
y = t[0] / 10000.0 # pazite kod /!
x = t[1]
len(y)
len(x)
xx = (x[0 : len(x)-1] + x[1 : len(x)]) / 2
len(xx)
close()
plot(xx, y)
xlabel('x')
ylabel('w(x)')
xlim(-5, 5)
xticks(linspace(-5, 5, 11))
sum(y)
help(savefig)
savefig('slika') # potrazite slika.png
savefig('slika.pdf') # potrazite slika.pdf
```

slika.pdf



Pylab, jedan script, dat.py

```
from pylab import *

deg = linspace(0, 4*360, 4*360 + 1)
wt = radians(deg)

f = exp(- wt / 2 / pi * 0.5) * sin(wt)
fp = exp(- wt / 2 / pi * 0.5)
fm = -fp

dat = array([deg, wt, f, fp, fm]).transpose()

np.save('deg.npy', deg)
np.save('f.npy', f)
np.save('fp.npy', fp)

np.savetxt('dat.txt', dat, fmt='%.4f')
```

Pylab, run, #1

na komandnoj liniji:

```
python dat.py
more dat.txt
less dat.txt
ls *.np*
ipython --pylab
```

Pylab, after the run

```
np.save?
np.savetxt?
np.load?
deg = np.load('deg.npy')
f = np.load('f.npy')
fp = np.load('fp.npy')
fm = -fp
plot(deg, f, 'r')
plot(deg, fp, 'b')
plot(deg, fm, 'b')
xlim(0, 360 * 4)
xticks(arange(0, 360 * 4 + 1, 180))
xlabel('wt [deg]')
ylabel('f(wt)')
savefig('datslik')
close()
```

Pylab, \LaTeX , estetika

```
help(rc)

close('all')
x = linspace(0, 1, 101)
y = x**2
rc('text', usetex = True)
rc('font', family = 'serif')
figure(0, figsize = (6, 4))
plot(x, y)
xlabel(r'$x$')
ylabel(r'$y$')
text(0.5, 0.4, r'$y = x^2$')
savefig('kvadrat.pdf', bbox_inches = 'tight')
```

Pylab, \LaTeX , „standardna“ „preambula“

```
rc('text', usetex = True)
rc('font', family = 'serif')
rc('font', size = 12)
rcParams['text.latex.preamble']=[r'\usepackage{amsmath}']
```

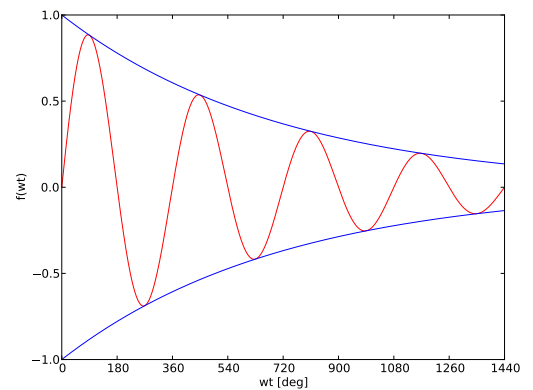
Pylab, run, #2 i #3

inside IPython:

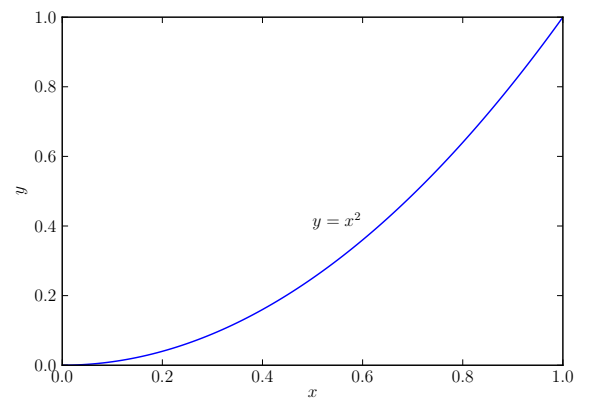
```
run dat.py
execfile('dat.py')
```

execfile radi i kod IDLE

datslik.pdf



kvadrat.pdf



Pylab, linear algebra, 0th part

hoću da rešim sistem jednačina:

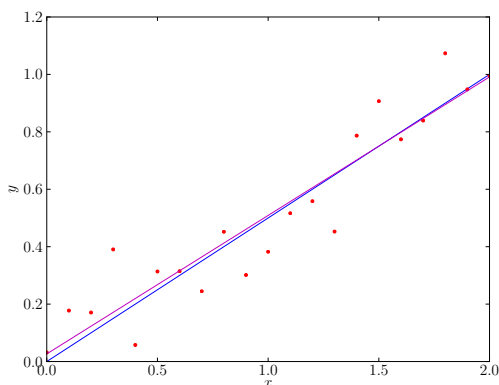
$$\begin{aligned}x + y &= 3 \\ x - y &= 1\end{aligned}$$

```
a = array([[1, 1], [1, -1]])
print a
b = array([[3], [1]])
print b
x = solve(a, b)
print x
b = array([3, 1])
print b
x = solve(a, b)
print x
```

Pylab, linear algebra, 1st part

```
randn?  
linalg.lstsq?  
  
x = linspace(0, 2, 201)  
y = x / 2  
  
xe = linspace(0, 2, 21)  
ye = xe / 2 + randn(21) * 0.1  
  
A = array([xe, ones(len(xe))]).transpose()  
t = linalg.lstsq(A, ye)  
print t  
type(t)  
len(t)  
a = t[0][0]  
b = t[0][1]
```

fitovanje.pdf



Pylab, linear algebra, 2nd part

```
close('all')  
  
rc('text', usetex=True)  
rc('font', family='serif', size='16')  
  
plot(x, y, 'b')  
plot(xe, ye, 'r.')  
  
yfit = a * x + b  
plot(x, yfit, 'm')  
xlabel(r'$x$')  
ylabel(r'$y$')  
savefig('fitovanje.pdf')
```

Šta je SymPy?

- ▶ paket za simboličko računanje koristeći Python
- ▶ cilj: isto što i wxMaxima
- ▶ ideja: Python sintaksa, poznata
- ▶ moduli, funkcije za simboličko računanje
- ▶ <http://sympy.org/en/index.html>
- ▶ <https://github.com/sympy/sympy/releases>
- ▶ <http://live.sympy.org/>
- ▶ uputstvo 0.7.6.1, 1995 strana, 03.09.2015.
- ▶ Ubuntu: Software Center ili Synaptic
- ▶ volite Mathematica sintaksu?
- ▶ odličan pregled i tutorial
- ▶ za one koji vole matematiku <http://www.sagemath.org/>

SymPy, počinjemo

- ▶ komandna linija, [isympy](#)
- ▶ stari poznanik, IPython, customized again
- ▶ pogledajte šta radi na početku, deklaracije symbols ...

SymPy, polinomi

```
p1 = x**2 + 2 * x + 1  
p1  
print p1  
pprint(p1)  
p2 = x + 1  
p2  
p3 = x**2 - 4  
p3  
p1 * p2 * p3  
_.expand()
```

SymPy, expand i apart

```
p1 / p2  
_.expand()  
p1 / p2  
_.apart()  
apart(p1 / p2, x)  
help(expand)  
help(apart)  
f = 1 / (x**2 * (x + 1))  
apart(f, x)  
together(_, x)  
_.expand()  
_.factor()  
_.subs(x, 3)
```

SymPy, opet problemi sa deljenjem

```
1 / 3  
Rational(1, 3)  
S('1/3')  
S('1 / 2')  
S('1/2') - S('1/3')
```

SymPy, konstante i izračunavanje

```
p2 = pi**2
p2
p2.evalf()
p2.evalf(100)
p2.n()
p2.n(200)
N(p2)
N(p2, 300)
exp(1)
e
E
E.evalf()
I**2
exp(I * pi) + 1
E**(I * pi) + 1
```

SymPy, realni brojevi

```
exp(I * x).expand(complex = True)
xr = Symbol('xr', real = True)
exp(I * xr).expand(complex = True)
```

SymPy, linearna algebra, opet isto ...

```
e1 = Eq(2 * x + y, 4)
e1
e2 = Eq(x - y, -1)
e1
e2
solve((e1, e2), (x, y))

e3 = Eq(x - y, -4)
e2
e3
solve((e2, e3), (x, y))

e4 = Eq(3 * x - 3 * y, -12)
e3
e4
solve((e3, e4), (x, y))
```

SymPy, ne baš linearna algebra

```
solve(p1, x)
solve(p2, x)
solve(p3, x)
solve(x**4 - 1, x)
solve(Eq(x**4, 1), x)
```

SymPy, Ramanujan

```
r = E**(pi * sqrt(163))
r
r.n(20)
r.n(21)
r.n(22)
r.n(23)
r.n(24)
r.n(25)
r.n(28)
r.n(30)
r.n(35)
r.n(500)
```

SymPy, linearna algebra

```
e1 = 2 * x + y - 4
e2 = x - y + 1
e1
e2
solve((e1, e2), (x, y))

e3 = x - y + 4
e2
e3
solve((e2, e3), (x, y))

e4 = 3 * x - 3 * y + 12
e3
e4
solve((e3, e4), (x, y))
```

SymPy, matrice, samo osnovno

```
A = Matrix([[x, 1], [1, y]])
A
print A
pprint(A)
A**2
A**(-1)
B = A.inv()
B
A**(-1) - B
A * B
simplify(A * B)
```

SymPy, limesi

```
f = sin(14 * x) / x
g = 1 / x
limit(f, x, 0)
limit(g, x, 0)
limit(g, x, 0, dir = '+')
limit(g, x, 0, dir = '-')
limit?
```

SymPy, izvodi

```
print p1
pprint(p1)
diff(p1, x)
diff(p1, x, 2)
diff(p1 * p2 * p3, x)
_.expand()
diff(p1 * p2 * p3, x, 14)
diff(p1*exp(3*x), x)
_.expand()
```

SymPy, Taylor

```
sin(x).series(x, 0, 10)
series(sin(x), x, 0, 10)
series?
series(sin(x), x, 1, 10)
series(exp(x), x, 0, 5)
series(exp(x), x, 5, 5)
source(series)
```

SymPy, diferencijalne jednačine

trigonometrijske funkcije kao rešenje

```
f(t).diff(t, 2) + 4 * f(t)
dsolve(_, f(t))
f(t).diff(t, 2) + 4 * f(t) - 24 * cos(4*t)
dsolve(_, f(t))
```

i eksponencijalne ...

```
f(x).diff(x, 2) - 4 * f(x)
dsolve(_, f(x))
Eq(f(x).diff(x, 2) - 4 * f(x), - 4 * exp(- 2 * x))
dsolve(_, f(x))
```

SymPy, diferencijalne jednačine, script, problem

Rešiti diferencijalnu jednačinu

$$\frac{d^2 f(t)}{dt^2} + 2 \frac{d f(t)}{dt} + 4 f(t) = 2 \sin(t)$$

i proveriti rešenje.

Napisati Python script koji rešava problem, prikazati jednačinu, rešenje i rezultat provere.

SymPy, integrali

```
integrate(p1, x)
integrate(p1, (x, 1, 2))
integrate(cos(x), x)
integrate(sin(x), (x, 0, pi))
```

SymPy, da obrišemo funkciju

```
f
del f
f = Function('f')
f
```

SymPy, diferencijalne jednačine, provera

```
deq = Eq(f(t).diff(t, 2) - 4 * f(t), - 4 * exp(-2*t))
deq
deq.lhs
deq.rhs
sol = dsolve(deq, f(t))
sol
sol.lhs
sol.rhs
s = sol.rhs
s
ver = deq.subs(f(t), s)
ver
ver = ver.doit()
ver
ver.lhs.expand()
ver.rhs
ver.expand()
```

SymPy, diferencijalne jednačine, script

```
from __future__ import division
from sympy import *

# cast
y, t = symbols('y t')
f = symbols('f', cls=Function)

# equation
deq = Eq(f(t).diff(t, 2) + 2 * f(t).diff(t, 1) + 4 * f(t), 2 * sin(t))
print
pprint(deq)

# solution
sol = dsolve(deq, f(t))
print
pprint(sol)

# verification
s = sol.rhs
ver = deq.subs(f(t), s)
ver = ver.doit()
ver = ver.expand()
print
pprint(ver)
```

- ▶ na kraju, a moglo je i na početku ...

Maxima	vs.	SymPy	objašnjenje
:	je	=	dodela vrednosti
is()	je	==	logički operator
=	je	Eq(*, *)	deklarisanje jednakosti

- ▶ očigledno postoji potreba za različitim =

in SymPy:

```
x**4 = 1      # error
x**4 == 1     # poređenje
Eq(x**4, 1)   # jednakost, jednačina
solve(_, x)   # koja može da se reši
```

$$4f(t) + 2\frac{d}{dt}f(t) + \frac{d^2}{dt^2}f(t) = 2\sin(t)$$

$$f(t) = \frac{\sqrt{C_1 \sin(\sqrt{3} t)} + C_2 \cos(\sqrt{3} t) e^{-t}}{13} + \frac{6 \sin(t) - 4 \cos(t)}{13}$$

True

PSAE, šta nismo uradili, a trebalo je, t << (potrebnog)

- ▶ **komandna linija**, <http://linuxcommand.org/tlcl.php>
- ▶ **regular expressions**, Charles Severance, video, 35', 23"
- ▶ detaljnije Code::Blocks ili neki drugi IDE
- ▶ **Eclipse** <http://www.eclipse.org/>
- ▶ ukratko LibreOffice, više pravila pisanja
- ▶ malo više vremena za Octave, ipak je jako dobar program
- ▶ makar malo vremena za **SciLab**, Xcos pre svega
- ▶ malo vremena za **GIMP**, mada nije problem
- ▶ malo vremena za **Inkscape**, ovo je veći problem
- ▶ **Qucs** <http://qucs.sourceforge.net/>
- ▶ **Ngspice** <http://ngspice.sourceforge.net/>
- ▶ **moj izbor**: julia i sage
- ▶ ovo je samo početak, mada je za sada ...

— K R A J —