

Osnove korištenja terminala

Leonard Volarić Horvat

Istraživački centar mladih
Fakultet elektrotehnike i računarstva

Sadržaj

- 1 Operacijski sustav
- 2 Terminal
 - Vrste sučelja
 - Terminal
- 3 Direktoriji i datoteke
- 4 Osnovne naredbe
 - Primjeri i zadaci
- 5 Upravljanje datotekama
- 6 Filteri, tokovi i cjevovodi
 - Preusmjerenje
 - Primjeri i zadaci
- 7 Skripte
 - Primjeri i zadaci
- 8 Package Manager
- 9 Demonstracija očitavanja sa senzora
- 10 Primjer mjerenja

Operacijski sustav

- Sučelje između hardvera i softvera
- Odvajanje detalja izvedbe hardvera od softvera
 - Windows sustav se koristi na isti način neovisno o fizičkom računalu
- korisnik → aplikacija → OS → uređaj

Operacijski sustav

Ilustracija: OS je sjemenka



- jezgra (*kernel*) - pristup resursima
 - upravljanje memorijom
 - sinkronizacija procesa
 - upravljanje vanjskim jedinicama - tipkovnica, miš, zvučnici...
 - ...
- ljuska (*shell*) - "ono što korisnik vidi"
 - **terminal**
 - *desktop environment*
 - aplikacije - uređivači teksta, web preglednici
 - ...

Terminal

Vrste sučelja

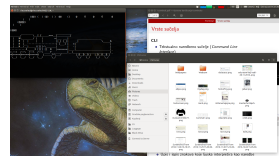
CLI

- Tekstualno naredbeno sučelje (*Command Line Interface*)
- Upis i ispis znakova koje ljska interpretira kao naredbe
- Isključivo tekstualni ulaz/izlaz, liniju po liniju
 - → **samo tipkovnica!**

```
auth.log.1          lightning      unattended-upgrades
auth.log.2.gz       nginx        upstart
auth.log.3.gz       php7.0-fpm.log vmware-installer
auth.log.4.gz       php7.0-fpm.log.10.gz cron
boot.log            php7.0-fpm.log.11.gz wtmp
boot-sav            php7.0-fpm.log.12.gz wtmp.1
bootstrap.log       php7.0-fpm.log.2.gz Xorg.0.log
btm                 php7.0-fpm.log.3.gz Xorg.0.log.old
btm.1               php7.0-fpm.log.4.gz Xorg.1.log
cups                php7.0-fpm.log.5.gz
dist-upgrade        php7.0-fpm.log.6.gz
dmesg               rincewind@rincewind-M551JK:~$ stat boot.log
File: 'boot.log'
Size: 4360      Blocks: 16      IO Block: 4096   regular file
Device: 8:0/2554 Inodes: 654897 Links: 1
Access: (0644/rw-r--r--)  Uid: (  0/   root)   Gid: (  0/   root)
Access: 2017-07-17 21:30:43.849033592 +0200
Modify: 2017-07-17 21:30:51.01303254 +0200
Change: 2017-07-17 21:30:51.01303254 +0200
Biting:
rincewind@rincewind-M551JK:~$ var/log$ echo "Hallo Welt!"
Hallo Welt!
rincewind@rincewind-M551JK:~$ var/log$
```

GUI

- Grafičko korisničko sučelje (*Graphical User Interface*)
- **Uvodi se koncept miša!**
- Bogatije, ali i kompleksnije rješenje



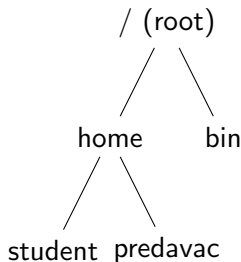
Terminal

- Osnovni CLI ulazno-izlazni program
 - Upis i ispis znakova koje ljuska interpretira kao naredbe
 - Osnovna sintaksa: `<ime_naredbe> [<zastavice>] [<argumenti>]`
-
- Nekad fizički uređaj za pristup ljusci
 - Danas program koji simulira fizički uređaj

Datoteke i direktoriji

Direktoriji

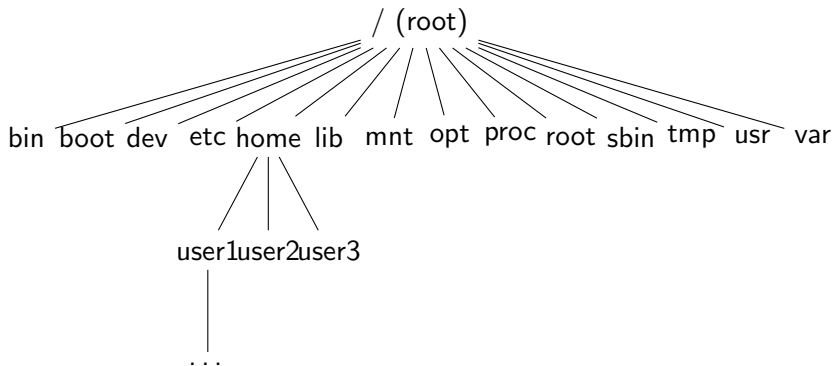
- Direktoriji su organizirani kao stablo (FHS - Filesystem Hierarchy Standard)
- U Unix operacijskom sustavu nema diskova
 - Sve je **jedno** stablo direktorija s **jednim** korijenom



Posebni direktoriji

- Svaki direktorij sadrži dva posebna direktorija
 - .. roditeljski direktorij (engl. *parent directory*)
 - . trenutni direktorij (engl. *current directory*)
- Primjeri
 - `ls .`
 - Ispisuje sadržaj trenutnog direktorija (uobičajeno `ls` ponašanje)
 - `ls ..`
 - Ispisuje sadržaj direktorija koji je roditelj trenutnom
- Koriste se za **relativno** adresiranje direktorija
- Dva oblika staze (*path*) do datoteke:
 - **apsolutna** staza: cijeli put od roota do ciljane datoteke
 - **relativna** staza: put od trenutnog direktorija do ciljane datoteke

Pregled direktorija sustava



Datoteke

- *Everything is a file!*
- Na Unix sustavima je sve neka vrsta datoteke
 - "obične" datoteke
 - direktoriji
 - blokovski uređaji - diskovi, USB stickovi
 - znakovni uređaji - zvučne i grafičke kartice, tipkovnice
 - mrežni socketi
 - ...
- Imenovanje:
 - u imenu ne smije biti znak /
 - datoteka se smatra skrivenom ako počinje točkom: .test
 - ekstenzija (.txt, .sh, .py, .java...) je tek konvencija i nije nužna za funkcionalnost

Osnovne naredbe

Osnovne naredbe za snalaženje po FHS

- **ls** - ispis sadržaja direktorija
- **pwd** - ispis trenutnog direktorija
- **cd** - promjena direktorija
- **ls** - ispis sadržaja direktorija
- **cat** - ispis sadržaja datoteke
- **touch** - stvaranje prazne datoteke
- **mkdir** - stvaranje novog direktorija
- **rm(dir)** - brisanje datoteke (direktorija)

- **man** - ispis detaljnih informacija o naredbi

Primjeri

- ❶ Pozicionirajte se u svoj direktorij Pictures
 - apsolutno: `cd /home/<username>/Pictures`
 - ili, ako smo u matičnom direktoriju, relativno: `cd Pictures`
- ❷ Ispišite trenutni direktorij
- ❸ Stvorite direktorij "Black Mesa" (s razmakom u imenu!)
 - `mkdir Black Mesa`
- ❹ Uđite u taj direktorij, uvjerite se da je prazan, pa se, koristeći relativnu stazu, pozicionirajte u svoj direktorij Music

Zadaci za vježbu

- 1 Pozicionirajte se u svoj direktorij Desktop
- 2 Stvorite direktorij Hitchhiker
- 3 U njemu stvorite datoteke Marvin, Hactar i .Magrathea (s točkom!)
- 4 Ispišite **cijeli** sadržaj direktorija
- 5 Koristeći relativnu stazu, pozicionirajte se u svoj direktorij Documents
- 6 **Jednom naredbom** (hint: man mkdir) napravite direktorij Verse i u njemu direktorij Serenity
- 7 Koristeći kratki izraz za matični direktorij (~), vratite se u njega i jednom naredbom obrišite direktorij Hitchhiker zajedno s njegovim sadržajem

Upravljanje datotekama

Upravljanje datotekama

Premještanje:

- **mv** - premještanje i preimenovanje datoteke
- **cp** - kopiranje datoteke

Podaci o datoteci:

- **file** - ispis tipa datoteke (po aplikaciji za pristup!)
- **ls -l <ime_datoteke>** - detaljni podaci o datoteci
- **stat** - (drugi) detaljni podaci o datoteci

Ispis:

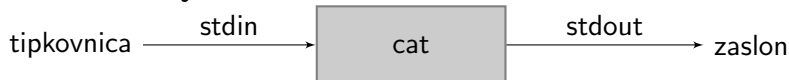
- **cat** - ispis cijele datoteke
- **less** - uredniji ispis cijele datoteke koji podržava listanje (*scrolling*)
- **head** - ispis prvih *n* linija datoteke
- **tail** - ispis zadnjih *n* linija datoteke
- **grep <riječ> <datoteka>** - traženje pojavljivanja riječi u datoteci

Filteri, tokovi i cjevovodi

- Svaki program na Linuxu ima definirane sljedeće ulaze i izlaze:
 - Standardni ulaz (stdin)
 - Standardni izlaz (stdout)
 - Standardni izlaz za greške (stderr)
- Svi su ti ulazi i izlazi vezani na terminal
 - Ako ih nismo preusmjerili uz pomoć specijalnih operatora

Primjer naredbe *cat*

- Grafička ilustracija ulaza i izlaza



- Naredba *cat* je filter!
 - Preuzima nešto na ulazu
 - Filtrira preuzete podatke
 - Prosljeđuje rezultat na standardni izlaz

Preusmjeravanje

Postoji nekoliko operatora za preusmjeravanje tokova:

- `<` - preusmjeravanje sadržaja datoteke na *input* naredbe
- `>` - preusmjeravanje *outputa* naredbe u datoteku
- `>>` - preusmjeravanje *outputa* naredbe na kraj datoteke
- `|` (*pipe*) - preusmjeravanje *outputa* naredbe na *input* iduće naredbe

Primjeri

- Napravite prazne datoteke *Wash* i *Zoe*
- Koristeći naredbu *echo*, upišite u datoteku *Wash* jednu liniju teksta
- Na isti način dodajte još jednu liniju, ali tako da očuvate i prethodni sadržaj
- Koristeći neki *text editor*, upišite u datoteku *Wash* 10 redova, a u *Zoe* 5 redova teksta
- Obje datoteke premjestite u direktorij *Verse/Serenity* iz prethodnog zadatka
- Ispišite prva 3 reda datoteke *Zoe*
- Koristeći opciju za negativan broj redaka ispišite prvih 7 redova datoteke *Wash*
- Ispišite druga 3 reda datoteke *Wash*

Zadaci

- Pozicionirajte se u direktorij `/usr/share/dict`
- Provjerite postoji li u datoteci `words` riječ "turtle" (hint: naredba `grep`!)
- Učinite isto za riječ "poslovanje"
- Provjerite koliko se puta u **drugih** 100 linija pojavljuje niz znakova "ga"

Skripte

Skripte

- Niz naredbi zapisan u tekstualnu datoteku
- Zapisane se naredbe jednostavno slijedno izvode
- Ime skripte obično završava na `.sh`
- Za pokretanje potrebno imati **dozvolu za izvršavanje** – `chmod +x skripta.sh`
- Pokretanje skripte: `./skripta.sh`

Primjer

```
#!/bin/bash
mkdir alpha_quadrant
cd alpha_quadrant
touch {betazed,earth,kronos,omicron_theta,vulcan}
echo "Before: "
ls -lh
echo "Live long and prosper" > vulcan
echo "After: "
ls -lh
echo "So long and thanks for all the fish!"
```

Zadatak

Napisati skriptu koja:

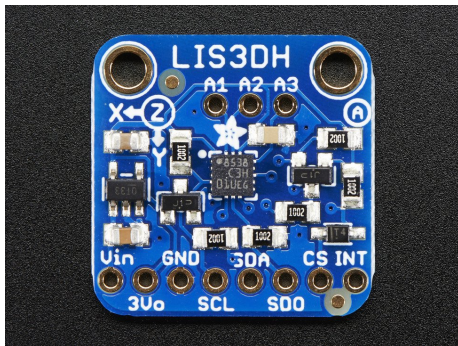
- Stvara direktorij `/tmp/my_boot` i pozicionira se u nju
- Iz izlaza naredbe `dmesg` uzima sve linije u kojima se pojavljuje riječ "boot" i sprema to u datoteku "original"
- Obrće poredak tih linija (koristeći prikladan oblik naredbe `sort`) i rezultat te operacije sprema u "reverse"
- Miče prvu liniju iz datoteke "original" i sprema tu promjenu
- Ispisuje detalje o sadržaju direktorija
- Pozdravlja korisnika porukom "Bok!"
- Briše direktorij `my_boot` i sav njegov sadržaj

Upravljanje paketima

- paket je program
- *yum, apt, pacman, zypper* ...
- Primjer naredbi za apt:
 - *apt update*
 - *apt install sl*
- Najčešće nužno koristiti administratorske ovlasti → **sudo**
- **sudo** - naredba koja privremeno daje administratorske ovlasti

Demonstracija očitavanja sa senzora

Akcelerometar - Adafruit LIS3DH



- Mjerenje na 3 osi
- 10-bitna preciznost
- Podrška za I²C i SPI protokole
- Vrlo mala potrošnja
- Prekidni pin
- Ulazi za ADC

Akcelerometar tvrtke Adafruit

Opis mjerenja

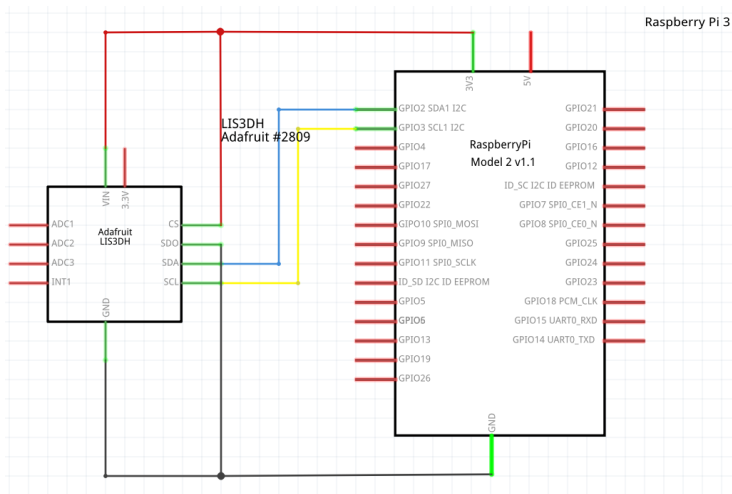
Korišten LIS3DH, spojen preko I²C-a

Mjerenja:

- ① Senzor pričvršćen za stol:
 - Kucanje
 - Udaranje
- ② Senzor pričvršćen za bas gitaru
 - Odsviran ton A (55Hz) - prigušeno titranje

Programska podrška: Python

Opis mjerenja

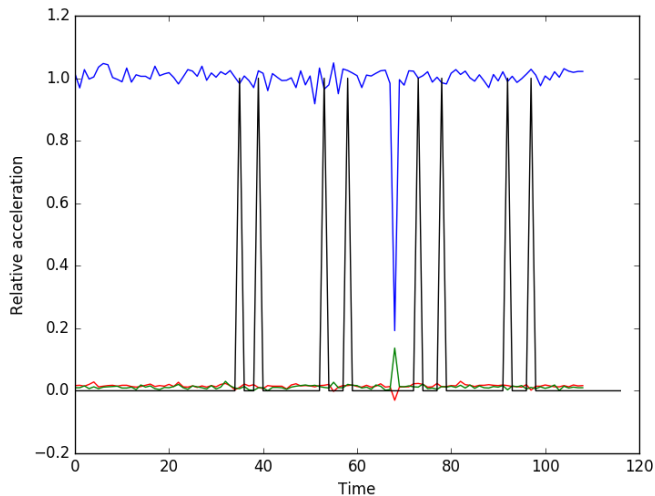


Programski kôd

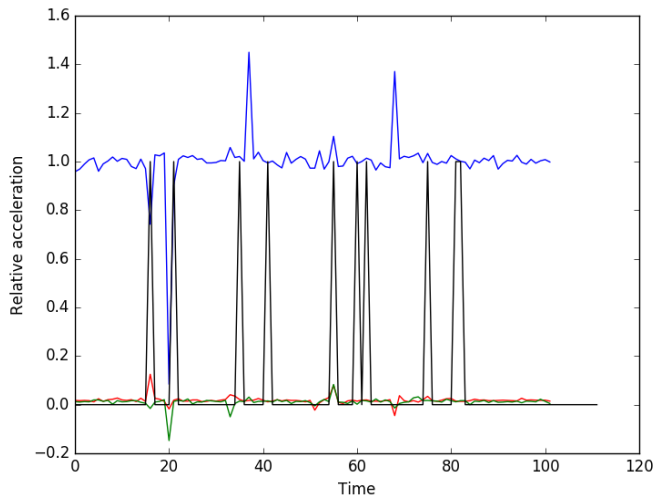
Čitanje podataka u Pythonu:

```
1 #!/usr/bin/python
2
3 from LIS3DH import LIS3DH
4 from time import sleep
5
6 if __name__ == '__main__':
7     sensor = LIS3DH(debug=True)
8     sensor.setRange(LIS3DH.RANGE_2G)
9     sensor.setClick(LIS3DH.CLK_SINGLE, 80, mycallback=clickcallback)
10
11     print "Starting stream"
12     while True:
13         x = sensor.getX()
14         y = sensor.getY()
15         z = sensor.getZ()
16
17 # raw values
18     print "\rX: %.6f\rY: %.6f\rZ: %.6f" % (x,y,z)
19     sleep(0.1)
```

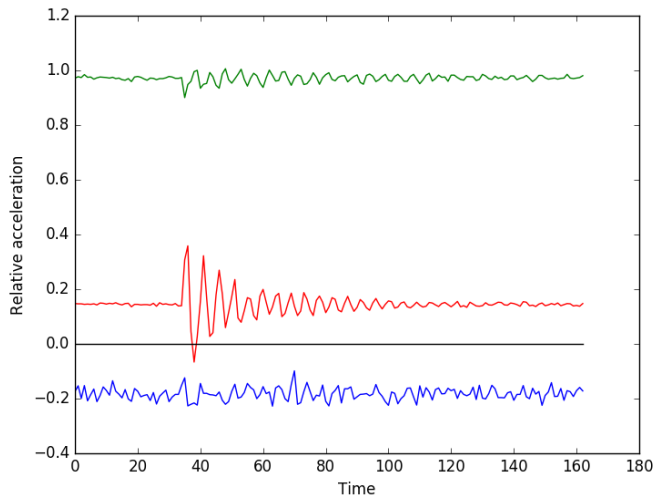
Primjer mjerenja - kucanje



Primjer mjerenja - udaranje



Primjer mjerenja - prigušeno titranje



Hvala na pažnji!