

# Operativni sustavi

Vježbe

Ljiljana Despalatović

Copyright @ 2017 Ljiljana Despalatović

Licensed under the Creative Commons Attribution 4.0 International (the "License"). You may not use this file except in compliance with the License. You may obtain a copy of the License at <https://creativecommons.org/licenses/by/4.0/>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

# Sadržaj

1	Uvod .....	5
1.1	Upute za vježbe	5
2	Vježba 1 .....	7
2.1	Instaliranje operativnog sustava Linux na virtualni stroj	7
3	Vježba 2 .....	9
3.1	Command line interface (CLI)	9
3.2	Naredbe Linux operativnog sustava	10
3.3	Korištenje repozitorija	11
3.4	Kompajliranje programa za Linux platformu	12
4	Vježba 3 .....	15
4.1	Naredbe za navigaciju datotečnim sustavom	15
4.2	Putanje	18

5	Vježba 4 .....	21
5.1	Wildcards	21
5.2	Naredbe za pretraživanje file systema	21
6	Vježba 5 .....	23
6.1	Redirekcija >, >>, <	23
6.2	Naredbe grep, cut, sort, head, tail	24
7	Vježba 6 .....	27
8	Vježba 7 .....	29
8.1	Dozvole i vlasništvo nad datotekama i direktorijima (file permission)	29
8.2	Shell skripte	32
9	Vježba 8 .....	35
10	Vježba 9 .....	39
10.1	Procesi	39
11	Vježba 10 .....	43

# Uvod

## Upute za vježbe

- Svaku vježbu treba predati tako da se izlaz naredbe `history` uploada na sustav Moodle.
- Datoteka s povijesti naredbi mora biti nazvana prema predlošku: `grupa_ime-prezime_vjezba`. Studenti s više imena ili prezimena svaku riječ trebaju odvojiti crticom, npr. `03_william-henry-bill-gates-iii-04`. Sva slova u nazivu datoteke su mala i bez „kvačica”. Broj grupe i broj vježbe uvijek su dvoznamenkasti, a popis grupa vidljiv je na sustavu Moodle.
- Primjer naziva datoteke (grupa 3, student Tux Linić, vježba 1): `03_tux-linic_01`.



# Vježba 1

## Instaliranje operativnog sustava Linux na virtualni stroj

Virtualni stroj (Virtual Machine, VM) je program koji simulira računalo i na koju možemo instalirati drugi operativni sustav. Operativni sustav (OS) računala zovemo host OS, a operativni sustav instaliran na VM zovemo guest OS.

U ovoj vježbi instalirat ćemo 32-bitnu Linux OS Mint distribuciju koja je bazirana na Ubuntu distribuciji koja je pak bazirana na Debian distribuciji. Sve Debian-based distribucije imaju isti sustav pakiranja i instaliranja programa.

**Zadatak 2.1** Slijediti sljedeće korake instalacije:

- Odabrati Linux distribuciju, te preuzeti njenu .iso datoteku.
- Kopirati .iso datoteku na disk.
- Pokrenuti Oracle VirtualBox manager.
- Iz izbornika odabrati Machine, New.
- Upisati karakteristike virtualnog stroja: Naziv, Tip (Linux), Verzija (Ubuntu 32-bit), RAM (1500 MB)
- Napraviti virtualni hard disk (VDI, Dynamic, 16 GB)
- Odabrati ISO sliku: VM settings, Storage, označiti DVD (Empty), Choose Virtual Optical Disk File
- Instalirati operativni sustav. Odabrati manualno određivanje veli-

čine particija.

- Prijedlog veličina particije
  - / ext4 12GB
  - swap linux swap max 1 GB
  - /home ext4 3GB
- username: student
- password: student

**Zadatak 2.2** Isprobati funkcionalnosti distribucije Linux Mint.

Postavite prozor virtualnog stroja preko cijelog ekrana (Full screen mode). Sve vježbe ubuduće radit ćemo u Linuxu namještenom preko cijelog ekrana. Pokrenite preglednik i pristupite sustavu Moodle. Isprobajte programe Calculator, File Browser, System monitor, xed ili gedit (editori teksta). Na kraju pokrenite terminal i isprobajte naredbe `echo` i `history`.

```
echo "Hello world"
```

```
echo "Ime Prezime"
```

```
history
```

Usmjerite izlaz naredbe `history` u datoteku koju ćete predati preko sustava Moodle (pažljivo pročitajte upute za vježbe):

```
history > naziv_datoteke
```

Pokrenite file manager, potražite stvorenu datoteku i pregledajte je u editoru teksta. Predajte datoteku na način opisan u uputama za vježbe.



## Vježba 2

### Command line interface (CLI)

Command line interface (CLI) je naziv za programe koji omogućuju unošenje naredbi u tekstualnom obliku koje potom izvršava Command line interpreter ili ljuska (engl. shell). Shell je sučelje između korisnika i operativnog sustava. Na Linux operativnim sustavima dostupno je nekoliko shellova, aktivni možemo detektirati naredbom `echo $SHELL`.

Shell je program koji interpretira naredbeni redak, parsira ga i pokreće procese navedene u njemu. Naredbeni redak ima odzivni znak (za `bash` je to znak `$` u korisničkom modu rada, a `#` u root modu). Parsiranje se vrši tako da se podijeli redak u riječi (engl. words) na sljedeći način:

- svako pojavljivanje navodnika označava tekst koji slijedi do sljedećeg navodnika kao jednu riječ
- svako pojavljivanje praznine označava kraj riječi
- naredbena linija završava novim redom ili znakovima `;`, `&`, `|`.

Svaka riječ u naredbenom retku je pridružena nizu stringova. Prva riječ ima indeks 0 u nizu, druga 1 itd.

Na Linux OS možemo koristiti programe `Konsole` ili `xterm` kao CLI ili instalirati neki drugi CLI (`yakuake`, `guake`, `terminator`, `eterm` itd.). Uobičajeno naziv za te programe je `konzola` ili `terminal`.

Kod korištenja terminala važan nam je podatak u kojem dijelu datotečnog sustava se nalazimo. Inicijalno je to home direktorij (u našem slučaju `/home/student`). Kratica za home direktorij je `~`.

### Primjer 3.1 Pozicioniranje u home direktorij

- `ls ~`
- `ls /home/student`

Osim home direktorija, bitan direktorij je root direktorij. Oznaka za njega je `/`.

### Primjer 3.2 Pozicioniranje u root direktorij

- `cd /`

## Naredbe Linux operativnog sustava

Opći oblik naredbi u Linux operativnim sustavima je:

```
$ command [option(s)][argument(s)]
```

Moguće opcije i argumente te njihov opis mogu se dobiti naredbom `man`.

```
man command
```

## Opće naredbe

### Zadatak 3.1 Izvršiti sljedeće naredbe u komandnoj liniji:

- Ispisati datum. Naredba `date`.
- Proučiti opcije naredbe `cal` koristeći naredbe `man` i `info`.
- Ispisati kalendar za 2020. Ispisati kalendar za drugi mjesec 2020. Ispisati kalendar za 1., 2. i 3. mjesec 2020. koristeći naredbu `cal`.
- Obrisati sadržaj ekrana koristeći naredbu `clear`.
- Ispisati poruku koristeći naredbu `echo`. Npr. `echo Hello world`.
- Ispisati fortune cookie. Naredba `fortune`. Ako program `fortune` nije instaliran, instalirati ga koristeći `sudo apt-get install fortune`.

## Korištenje naredbe `history`

### Zadatak 3.2 Izvršiti sljedeće naredbe u komandnoj liniji.

- history
- history 10
- 10!
- ponovno pokretanje naredbe iz historyja s modifikacijom: echo:s/world/you!
- pretraga history-ja: <Ctrl-R>ech

## Tips and tricks

**Zadatak 3.3** Isprobati sljedeće u komandnoj liniji:

- Pomoću strelice gore i strelice dolje možemo pristupiti već izvršenim naredbama. Isprobati!
- Za pregled prethodnih naredbi i rezultata koristiti shift+page-up, shift+page-down.
- Naredbe i putanje se mogu dovršiti korištenjem tipke <TAB>. Isprobajte: his<TAB>
- Selektirajte neku od predhodno napisanih naredbi i kopirajte je koristeći srednji botun miša ili istovremeno oba botuna na touchpadu. Selekcija - copy, srednji botun - paste.

## Izvršavanje naredbi s povišenom razinom ovlasti

Korisnici grupe sudoers mogu pokretati naredbe s povišenom razinom ovlasti:

- sudo naredba
- sudo su, lozinka je ista (student).
  - Uočite drugačiji komandni prompt.
  - Izadite sa exit.

## Korištenje repozitorija

Svaka distribucija Linuxa dolazi sa svojim repozitorijem tj. kolekcijom programa kompajliranih za tu distribuciju i verziju Linuxa. Repozitorij se nalazi na serveru, a lokalno je pohranjen popis dostupnih programa. Osim glavnog repozitorija, korisnik može dodati i druge repozitorije. Programi u repozitoriju dostupni su u obliku paketa (engl. package). Lokalno, na računalima, je pohranjena lista dostupnih paketa. Listu dostupnih paketa

osvježavamo naredbom `apt-get update` pokrenutom kao `root`.

Za rad sa softverskim paketima (pretraga i instaliranje) može se koristiti GUI aplikacija ili se može koristiti naredbena linija. Programi za baratanje paketima zovemo Package Managerima. Za Ubuntu, Mint, Debian i ostale distribucije izvedene iz Debian distribucije GUI programi su `synaptic`, `Software Centar` ili `Software Manager` te `Update Manager`. Program za rad iz naredbenog retka je `APT` (`Advanced Package Tool`).

Osnovne `apt` naredbe:

- `apt-get update` - osvježava listu dostupnih paketa (pokrenuti kao `root`)
- `apt-cache search name` - pretraga repozitorija, vraća sve pakete koji u svom imenu ili opisu imaju riječ `name`
- `apt-get install paket` - instalacija paketa na računalo (pokrenuti kao `root`).

## Instaliranje programa (Ubuntu, Mint i ostale Debian distribucije)

**Zadatak 3.4** Izvršiti sljedeće naredbe u komandnoj liniji:

- Pokreniti program `synaptic` (kao `root`) i instalirajte program `code::blocks` ili neki drugi C IDE. Zatvorite `synaptic`.
- Koristeći naredbu `apt-get` napravite update repozitorija (`sudo apt-get update`), a zatim instalirajte neki program (npr. `sudo apt-get install fslint`).
- Pronađi u repozitoriju programe za pisanje nota `apt-cache search "music score"` i instalirajte ga.

## Kompajliranje programa za Linux platformu

Programi za Linux OS u najvećoj mjeri su pisani u programskim jezicima C/C++, Python, Perl i Java. Među njima, programi napisani u C/C++ se kompajliraju te ih je uglavnom moguće preuzeti u obliku izvornog koda, te samostalno kompajlirati. Sa većinom distribucija Linux dolazi i C/C++ kompajler GNU GCC (GNU Compiler Collection) no ako već nije instaliran može se instalirati zajedno sa ostalim potrebnim alatima i bibliotekama koristeći (za distribucije izvedene iz Debiana) `sudo apt-get install build-essential`. To je ustvari kolekcija alata za C, C++, Objective-C, Fortran,

Ada i Go jezike uključujući i biblioteke za te jezike. GNU GCC izdan je pod GNU licencom slobodnog softvera.

## C kompajler

**Zadatak 3.5** Kompajlirati jednostavan C program.

- Kreirati datoteku `hello.c` koristeći naredbu `touch`.
- Otvoriti datoteku programom `xed`, `gedit`, `kate`, `nano`, `codeblocks` ili nekim editorom.
- Napisati neki jednostavan C program (npr. program koji ispisuje poruku "Hello world!").
- Iskompajlirati program: `gcc hello.c -o hello`. Opcija `-o` daje ime izvršnoj datoteci. Ako te opcije nema, izvršna datoteka zvat će se `a.out`
  - Ako nedostaje standardna biblioteka instalirajte je `sudo apt-get install libc6-dev`.
  - Ako nedostaje kompajler možete ga instalirati zajedno sa pripadajućim paketima `sudo apt-get install build-essential`.
- Pokreniti izvršnu datoteku: `./hello`

Izvršna datoteka je binarna datoteka. Takvim datotekama barata OS. Binarne datoteke su datoteke koje ne sadrže samo tekst. Vrstu datoteke možemo detektirati naredbom `file`. Za izvršnu datoteku troslovnna oznaka dobivena naredbom `file` bit će ELF, slika spremljena kao `png` imat će oznaku `PNG`, dok će `pdf` dokument imati oznaku `PDF`. U Linux zajednici uobičajeni je naziv `binaries` za izvršne binarne datoteke.

**Zadatak 3.6** Sadržaj binarne datoteke `hello` otvorite programom `hexdump` ili `od`.

**Zadatak 3.7** Naredbom `strings` ispišite sve stringove u binarnoj datoteci i zapišite ih u datoteku `vjezba1bin.zip`. Naredbom `file` provjerite format datoteke `vjezba1bin.zip`.

**Zadatak 3.8** Koje informacije o datoteci možete dobiti naredbom `stat`?



## Vježba 3

### Naredbe za navigaciju datotečnim sustavom

Putanja do datoteke ili direktorija je niz imena direktorija odvojenih delimiterom '/' koji jednoznačno vode do lokacije na kojoj se datoteka ili direktorij nalaze. Putanja do datoteke zajedno sa imenom datoteke čini pathname.

Putanje mogu biti apsolutne ili relativne. Apsolutna putanja počinje od korjenskog direktorija '/', a relativna od tekućeg direktorija.

Datotečni sustav ima hijerarhijsku strukturu direktorija. Na vrhu je root direktorij (/), njegova djeca su bin, home, sbin, var, proc itd. Unutar home direktorija nalaze se korisnički direktoriji (npr. student) i unutar njega, korisničke datoteke.

- Naredba `pwd` ispisuje radni ili tekući direktorij. Prilikom pokretanja ljuske (eng. shell), tekući direktorij biti će `/home/student`.
- Naredbom `cd` mijenja se tekući direktorij (change directory).

```
cd path
```

Specijalne oznake ili kratice za rad sa direktorijima su dane u tablici.

znak	značenje	primjer
.	tekući direktorij	<code>ls .</code>
..	roditeljski direktorij	<code>ls ..</code>
~	home direktorij	<code>ls ~</code>

#### Primjer 4.1 Korištenje naredbe cd.

- cd naziv\_dir - naziv\_dir postaje tekući direktorij
  - cd .. - prebacivanje u roditeljski direktorij
  - cd /var/spool/anacron - /var/spool/anacron postaje tekući direktorij
  - cd ~ prebacivanje u korisnikov home direktorij
- Ljuska proširuje znak tilda u putanju do korisnikovog home direktorija

```
echo ~
```

- ls - daje popis datoteka i direktorija u danom direktoriju (ako nema parametara, izlista se tekući direktorij)

```
ls direktorij
```

#### Primjer 4.2 Primjer korištenja naredbe ls.

- ls naziv\_dir - sadržaj direktorija naziv\_dir
- ls .. - sadržaj roditeljskog direktorija
- ls /var - sadržaj direktorija /var

Neke od opcija naredbe ls

- ls -l prikazuje detaljniji ispis
  - ls -a prikazuje i datoteke koje počinju s . (hidden file)
  - ls -al kombinacija prethodne dvije opcije
  - ls -r ispisuje sadržaj direktorija u obrnutom redoslijedu (leksikografski)
- mkdir kreira direktorij

```
mkdir naziv_dir
```

- rmdir briše direktorij. Direktorij treba biti prazan, inače rm -r

```
rmdir naziv_dir
```

- cp kopira datoteke ili direktorije

```
cp sto gdje
```



**Primjer 4.3** Primjer korištenja naredbe `cp`

- `cp naziv.txt ~/dir1/` - kopira datoteku `naziv.txt` iz tekućeg direktorija u direktorij `~/dir1/`
  - `cp naziv.txt ../` - kopira datoteku `naziv.txt` iz tekućeg direktorija u roditeljski direktorij `..`
  - `cp naziv.txt ../naziv2.txt` - kopira datoteku `naziv.txt` iz tekućeg direktorija u roditeljski direktorij `..` s novim nazivom.
  - `cp -R dir1 dir2` - kopira direktorij i sve njegove elemente u drugi direktorij
- 
- `mv` premješta datoteke ili direktorije

```
mv sto gdje
```

**Primjer 4.4** Primjer korištenja naredbe `mv`.

- `mv naziv.txt ~/dir1/` - premješta datoteku `naziv.txt` iz tekućeg direktorija u direktorij `~/dir1/`
- `mv -R dir1 dir2` - premješta direktorij i sve njegove elemente u drugi direktorij
- `mv naziv.txt ../naziv2.txt` - premješta datoteku `naziv.txt` iz tekućeg direktorija u roditeljski direktorij `..` s novim nazivom.
- `mv -R dir1 dir2` - premješta direktorij i sve njegove elemente u drugi direktorij

- Zadatak 4.1**
- Isprobajte što rade sljedeće naredbe: `cd`, `ls`, `mkdir`, `rmdir`, `pwd`.
  - Koristeći naredbu `touch` kreirati novu datoteku.
  - Sadržaj nekog direktorija ispišite u datoteku `listing` koristeći preusmjerenje `>`.
  - Koristeći naredbe `cat` i `less` prikazati sadržaj neke datoteke.

**Zadatak 4.2**

Napravite praznu datoteku naziva `myfile`.

- Zadatak 4.3** Riješite sljedeće zadatke koristeći naredbe `cd`, `ls`, `mkdir`, `rmdir`, `mv`, `cp`, `pwd`:

- Ispitajte koji je tekući direktorij (direktorij u kojem se nalazite).

Naredba `pwd`.

- Promjenite tekući direktorij u root (`/`). Naredba `cd`.
- Ispišite putanju do svog home direktorija. Naredba `echo $HOME`.

**Zadatak 4.4** · Prebacite se u svoj home direktorij.

- Napravite direktorij `mydir` i prebacite se u njega.
- Napravite poddirektorij `tmp` i u njemu datoteku `prva`.

Rješenja zadatka u kojima je potpis napisan u zaglavlju provjeravaju se nakon što su napravljeni svi pojedinačni koraci u zadatku.

**Zadatak 4.5** Prebacite se u svoj home direktorij.

- Iscrtajte strukturu direktorija naredbom `tree`.
- Izlistajte popis datoteka u tekućem direktoriju. Naredba `ls`.
- Ispitajte opcije naredbe `ls`: `ls -a`, `ls -l`, `ls -d`, `ls -t`, `ls ..`, `ls .` i kombinacije npr. `ls -ltr`.
- Kopirajte datoteku `myfile` u novi direktorij `copies`. Naredba `cp`.
- Napravite novu datoteku `test` i preimenujte je u `test.bak` koristeći naredbu `mv`.
- Pomaknite datoteku `test.bak` u direktorij `copies`.
- Izlistajte sadržaj direktorija `/usr` u obrnutom redoslijedu.
- Kopirajte direktorij `mydir` i sve datoteke u njemu u direktorij `mydir2`. Koristite opciju `cp -R`.

## Putanje

Da bi pokrenili program iz komandne linije trebamo navesti putanju do izvršne datoteke programa. Kako bi se skratio taj postupak, pogotovo za programe koje često pokrećemo, možemo putanje do programa dodati u specijalnu varijablu `PATH`. Na taj način, sve programe koji se nalaze u direktorijima navedenim u `PATH` varijabli, možemo pokretati navodeći samo njihovo ime.

Varijabla `PATH` je jedna od varijabli okruženja (engl. environment variable). Popis svih varijabli okruženja može se dobiti naredbom `env`.

- Prikažite direktorije koji se nalaze u varijabli `PATH`. Naredba `echo $PATH`
- Ispišite sve varijable okruženja.

Dodati ili promijeniti sadržaj varijable PATH možemo privremeno, za tekuću sesiju, ili stalno. Privremeno ćemo to napraviti naredbom `export`, a stalno promjenom u datoteci `~/.bash_profile`.



## Vježba 4

### Wildcards

Wildcards je skup simbola koje ljska zamjenjuje za grupu znakova u stringu prema dogovorenim pravilima.

znak	značenje	primjer
*	zamjenjuju grupu znakova bilo koje veličine	echo D*
?	zamjenjuje jedan znak	echo ?ocument
[]	lista mogućih znakova	ls *[tT]
-	raspon znakova	ls *[1-5]
!	invertira listu znakova	ls *![1-5]

**Zadatak 5.1** Ispišite sve datoteke u direktoriju `/usr/bin` koje:

- završavaju sa `config`.
- počinju sa `A` ili `a`.
- završavaju sa znamenkom.
- imaju znamenku `1` na drugoj poziciji.
- nemaju znamenku na zadnjoj poziciji.

### Naredbe za pretraživanje file systema

Pretraživanje naredbom `find`

Sintaksa `find <path> -name <searchstring>`.

opcija	opis	primjer
-name	pretraživanje po imenu	find /usr/bin/ -name 'v*'
-iname	pretraživanje po imenu (case insensitive)	find /usr/bin/ -iname 'v*'
-type	pretraživanje po tipu	find / -type d -iname '*s'
-mtime	pretraživanje datoteka modificiranih u zadanom vremenu	find /var/backups -mtime +5
-size	pretraživanje po veličini u blokovima	find /var/backups -size +50
-newer	pretraživanje datoteka novijih od navedene	find /var/backups -newer /var/lock

## Pretraživanje naredbom locate

Naredba `locate` omogućava brzo pronalaženje indeksiranih datoteka u datotečnom sustavu. Primjer: `locate password`. Indeksiranje se vrši naredbom `updatedb`.

**Zadatak 5.2** Riješite sljedeće zadatke koristeći `find` ili `locate` naredbu:

1. Nađite sve datoteke u `/usr/games` direktoriju čija imena počinju malim slovom.
2. Nađite sve datoteke u `home` direktoriju kreirane danas.
3. Nađite sve datoteke u `home` direktoriju starije od tjedan dana.
4. Nađite lokaciju programa `rmdir`.
5. Nađite lokaciju datoteke `.bash_history`. Što je u njoj?
6. Pronađite datoteku `.bash_logout` naredbom `locate`.
7. Pronađite sve datoteke u `home` direktoriju veće od 5MB. Naredba `find . -size +5000k`.
8. Pronađite sve datoteke na file systemu koje sadrže u imenu riječ `kernel`. Naredba `find / -iname *kernel*`

## Vježba 5

### Redirekcija `>`, `>>`, `<`

Programi koje pokrećemo iz komandne linije ispisuju rezultate u terminalu. Također, ako dođe do greške, poruke o greški se ispisuju u terminalu. Ako program ima ulazne podatke, oni se upisuju putem terminala kao argumenti komandne linije. Svaki program ima pridružena tri streama: standardni ulaz STDIN (0), standardni izlaz STDOUT (1) i standardna greška STDERR (2).

Ako ne želimo da izlaz iz programa ili ulaz u program idu na standardni stream, možemo to promijeniti. To se zove redirekcija. Operatori redirekcije sa primjerima dani su u sljedećoj tablici.

operator	opis	primjer
<code>&gt;</code>	dodavanje u datoteku ili stdout	<code>ls ~ &gt; lista.dat</code>
<code>&gt;&gt;</code>	dodavanje na kraj datoteke	<code>ls -al ~ &gt;&gt; lista.dat</code>
<code>&lt;</code>	ulaz iz datoteke ili stdin	<code>wc -l &lt; lista.dat</code>
<code>2&gt;</code>	greške (2-stderr) se ispisuju u datoteku	<code>find /etc/ -iname gre* 2&gt; err.dat</code>

- Nakon svakog primjera pogledati datoteke u koje je izlaz preusmjeren.
- Pogledati `man wc`.
- `/dev/null` je specijalni file u koji se mogu preusmjeriti neželjeni podaci. `/dev/null` je uvijek prazan.

**Zadatak 6.1** Redirekcija:

1. Izvršite sljedeće naredbe:

- `find /etc/ -iname gr*`
- `find /etc/ -iname gr* 2> vjezba5/err.dat`
- `find /etc/ -iname gr* 2> /dev/null`

Koja je razlika između gornjih naredbi?

2. Sadržaj direktorija `/etc/init` spremite u datoteku `etc.dat` (u direktorij `vjezba5`).

3. Koristeći redirekciju i program `wc` prebrojite koliko bajtova ima datoteka `etc.dat`.

4. Provjerite rezultat uspoređujući ga sa ispisom `ls -al etc.dat`.

5. Izvršite sljedeće naredbe:

- `cat < etc.dat`
- `cat < etc.dat > etccopy.dat`
- `cat < etc.dat > etccopy.dat 2> error.dat`

Koja je razlika između gornjih naredbi?

## Naredbe `grep`, `cut`, `sort`, `head`, `tail`

naredba	opis	primjer
<code>cut</code>	ispisuje određene stupce odvojene delimiterima iz datoteke	<code>cut -f1 -d. etc.dat</code>
<code>grep</code>	ispisuje linije u kojima se pojavljuje pattern	<code>grep hal etc.dat</code>
<code>sort</code>	sortira linije u tekstualnoj datoteci	<code>sort etc.dat</code>
<code>head</code>	ispisuje prvih $n$ linija datoteke (default je 10)	<code>head etc.dat</code>
<code>tail</code>	ispisuje zadnjih $n$ linija datoteke (default je 10)	<code>tail etc.dat</code>

**Zadatak 6.2** Pogledajte sadržaj datoteke `/etc/passwd` (koristiti naredbu `cat` ili `less`).

1. Ispisati samo korisničko ime i home direktorij (tj. prvo i šesto polje) u datoteci `/etc/passwd`.

```
cut -f1,6 -d: /etc/passwd
```

`-d` određuje koji je delimiter koji odvaja polja u redku (u našem primjeru je `:`)

`-f` određuje koja polja će se prikazati (u našem primjeru prvo i šesto)



2. Iz datoteke `/etc/passwd` ispisati samo redak koji sadrži riječ `root`.  
Koristiti naredbu `grep`
3. Ispisati sortiranu datoteku `/etc/passwd` koristeći naredbu `sort`.
4. Ispisati prvih 20 linija `/etc/passwd` datoteke. Koristiti naredbu `head`.
5. Ispisati zadnjih 10 linija `/etc/passwd` datoteke. Koristiti naredbu `tail`.



## Vježba 6

### Pipeline

Pipe ili pipeline je niz od dvije ili više naredbi odvojenih vertikalnom crtom |. Standardni izlaz prve naredbe, standardni je ulaz druge naredbe.

**Primjer 7.1** Izvršite sljedeće naredbe:

- `ls | wc -l`
- `ls | grep dat`
- `ls | tac -` ispis u obratnom redoslijedu (pogledati `man tac`)

**Zadatak 7.1** Napravite direktorij `misc` i pozicionirajte se u njega. Riješite sljedeće zadatke:

1. Snimite tri rečenice u tri različite datoteke naziva `prva`, `druga` i `treca` (izlaz naredbe `echo` usmjerite u datoteke).
2. Ispišite na terminal spojeni sadržaj sve tri datoteke: `cat prva druga treca`.
3. Spojite sadržaj svih datoteka u datoteku naziva `prva_druga_treca`.
4. Sortirani sadržaj spojene datoteke ispišite na terminal (naredba `sort`).
5. Spojite sadržaj svih pojedinačnih datoteka sortiran po abecedi u datoteku naziva `prva_druga_treca_sorted`
6. Prve dvije linije sortirane datoteke ispišite na terminal (naredba `head`).
7. Zadnje dvije linije sortirane datoteke ispišite na terminal (naredba

tail).

8. Kombinirajući naredbe `head` i `tail` ispišite samo drugu liniju sortirane datoteke.

**Primjer 7.2** Izvršite sljedeće naredbe:

- `ls | grep prva | wc -L`
- `ls -l | sort -k5 -r` (sortiranje po petom stupcu abecedno, obrnuto)
- `ls -l | sort -k5 -n` (sortiranje po petom stupcu, numerički)

**Zadatak 7.2** Riješite sljedeće zadatke:

- Izlistajte datoteke i direktorije u direktoriju `misc` sortirane po veličini.
- Iz datoteke `/etc/passwd` izdvojite prvi i šesti stupac koji se odnose na korisnika `mail`. Kombinirajte naredbe `grep` i `cut`.

**Zadatak 7.3** Dana je datoteka `vjezba06.csv` u kojoj su rezultati ispita. Datoteku treba preuzeti sa Moodla. Svaki redak sadrži ime, prezime, matični broj i bodove. Zbog GDPR-a potrebno je rezultate objaviti bez imena i prezimena, poredane po broju bodova. Napravite sljedeće korake, povezujući naredbe pipelineom:

- Ispišite matične brojeve i bodove na terminal (naredba `cut`, sa delimiterom `space`, tj. znakom razmaka koji se koristi ovako: `-d' '`). Ponovite naredbu i njen izlaz upišite u datoteku `mb_list`.
- Sortirajte studente i rezultate po abecedi (prikažite samo prezime i broj bodova) i upišite ih u datoteku `mb_sort`.
- Izdvojite tri najbolje napisana kolokvija (cijeli redak iz datoteke `vjezba06.csv`): ispis naredbe `cat` sortirajte po bodovima, a naredbom `head` izdvojite potrebne retke. Rezultat upišite u datoteku `top_list`.
- Iz datoteke `top_list` izvucite matični broj trećeg najboljeg studenta i ispišite ga na terminal.

## Vježba 7

### Dozvole i vlasništvo nad datotekama i direktorijima (file permission)

#### Korisnici i grupe

Da bi se datoteke zaštitile od neovlaštenog pristupa, Linux dozvoljava postavljanje dozvola korištenja datoteka i direktorija. Dozvole se određuju za:

- vlasnika datoteke (user)
- članove grupe kojoj je datoteka dodjeljena (group)
- sve ostale korisnike (other).

**Primjer 8.1** Ispišite sadržaj tekućeg direktorija naredbom `ls -l`. Prvih 10 znakova označava dozvole korištenja datoteke, slijedi broj koji označava broj linkova, a nakon toga polja koja označavaju vlasnika, grupu, veličinu, vrijeme zadnje promjene i naziv datoteke.

```
-rwxr-xr-x 1 os os 10103 2010-12-15 10:58 main
-rw-r--r-- 1 os os 248 2010-12-15 10:58 main.c
```

U gornjim primjerima:

- Prvi znak (- ili d) označava tip datoteke (datoteka ili direktorij).
- Sljedeća tri znaka (rwx ili rw-) označavaju dozvole vlasnika datoteke.
- Sljedeća tri znaka (r-x ili r--) označavaju dozvole za grupu.

- Sljedeća tri znaka (r-x ili r--) označavaju dozvole za ostale korisnike.

Za datoteke:

Troslovna kombinacija slova r, w i x označavaju redom: pravo čitanja, pisanja i izvršavanja datoteke.

Za direktorije:

- r: korisnik može vidjeti sadržaj direktorija (npr. s naredbom ls).
- w: korisnik može mijenjati sadržaj direktorija tj. kreirati, brisati i preimenovati datoteke u direktoriju.
- x: korisnik može koristiti direktorij kao svoj tekući direktorij, tj. može ući u njega naredbom cd.

**Zadatak 8.1** Nakon svake izvršene naredbe u primjerima i zadacima, izvršite naredbu `ls -l` kako bi vidjeli promjene.

- Napravite praznu datoteku `greeting`.
- Provjerite prava koja su postavljena nad datotekom
- Napravite direktorij `backup`
- Provjerite prava koja su postavljena nad direktorijem

## Mijenjanje dozvola

Dozvole se mogu promijeniti naredbom `chmod`. Sintaksa:

```
chmod <MODE> <naziv_datoteke>
```

<MODE> može biti napisan simboličkom ili oktalnom sintaksom.

### Simbolička sintaksa

```
chmod <TKO OPERATOR ST0> <naziv_datoteke>
```

<TKO> može biti:

- u korisnik
- g grupa
- o ostali
- a svi

<OPERATOR> može biti:

	u g o		
	754		
	/   \		
access	r w x	r w x	r w x
binary	4 2 1	4 2 1	4 2 1
enabled	1 1 1	1 0 1	1 0 0
result	4 2 1	4 0 1	4 0 0
total	7	5	4

- + dodavanje prava
- - oduzimanje prava
- = skidanje svih prava i dodavanje specificiranih

<ST0> može biti r, w i x.

**Primjer 8.2** Primjer simboličkog načina pridjeljivanja prava korištenja:

- `chmod u=rw,go=r kopija.c` - korisnik dobija pravo pisanja i čitanja, a grupa i ostali samo čitanja.
- `chmod a+x main` - svi dobiju pravo izvršavanja uz već postojeća prava.

**Zadatak 8.2** Nakon svake izvršene naredbe u primjerima i zadacima, izvršite naredbu `ls -l` kako bi vidjeli promjene.

- Promijenite prava nad datotekom `greeting` tako da svi dobiju pravo izvršavanja uz već postojeća prava
- Promijenite prava nad datotekom `greeting` tako da svi dobiju sva prava
- Promijenite prava nad direktorijem `backup` tako da grupi skinete sva prava

Oktalna sintaksa

```
chmod <ZZZ> <naziv_datoteke>
```

**Primjer 8.3** Primjer oktalnog načina pridjeljivanja prava korištenja:

- `chmod 644 kopija.c` - korisnik dobija pravo pisanja i čitanja, a grupa i ostali samo čitanja.
- `chmod 755 main` - svi dobiju pravo izvršavanja uz već postojeća prava.

**Zadatak 8.3** Nakon svake izvršene naredbe u primjerima i zadacima, izvršite naredbu `ls -l` kako bi vidjeli promjene.

- Kreirajte datoteku `main.c` s jednostavnim programom u c-u.
- Iskompajlirajte ga: `gcc main.c -o main`
- Pokrenite program: `./main`
- Napravite kopiju datoteke `main.c`. Neka se kopija zove `kopija.c`.

**Zadatak 8.4** Datoteci `kopija.c` dodijelite takva prava da samo user može mijenjati datoteku. Napravite manju izmjenu u datoteci i snimite je.

**Zadatak 8.5** Zabranite svima čitanje, pisanje i izvršavanje datoteke `main.c`. Pokušajte prikazati njen sadržaj na ekranu. Pokušajte izmijeniti sadržaj datoteke.

**Zadatak 8.6** Korisniku skinite prava čitanja i pisanja za datoteku `main`. Pokušajte pokrenuti program.

**Zadatak 8.7**

- U datoteku `greeting` upišite riječ `hello`.
- Kopirajte datoteku `greeting` u direktorij `backup`.
- Svim korisnicima skinite sva prava nad datotekom `greeting`.
- Pokušajte kopirati datoteku `greeting` u direktorij `backup`.

## Shell skripte

Skripta je datoteka koja sadrži naredbe ljuske (engl. shell). Ljuska čita datoteku i izvršava naredbu po naredbu kao da je pisana putem tipkovnice. Na taj način ne moramo pamtititi naredbe, poslovi se mogu izvršavati u određeno vrijeme, ne gubimo vrijeme rješavajući problem putem GUI-a.

### Pisanje shell skripte

Koraci:

1. Pokrenuti neki editor (npr. `gedit` ili `nano`)
2. U prvom retku napisati `#!/bin/bash`
3. Upisati naredbe i spremiti skriptu pod nekim nazivom (npr. `skripta.sh`). Nastavak datoteke nije obavezan, ali `sh` je uobičajen.
4. Skripti treba dodijeliti pravo izvršavanja (x) npr. `chmod 755 script.sh`



5. Skripta se pokreće tako da se upiše apsolutna ili skraćena putanja do nje, npr. `./script.sh`

**Zadatak 8.8** Napišite skriptu Hello world i izvršite je.

```
#!/bin/bash  
echo "Hello world!"
```



## Vježba 8

### Shell skripte

#### Elementi shell skripte

- Varijable se definiraju ovako: `X="Hello"`, a vrijednost se dohvaća sa `$X`. U varijablu možemo spremiti i vrijednost nekog aritmetičkog izraza koristeći dvostruke okrugle zagrade npr. `X=$((Y+Z))`
- Standardne (environment) varijable: su varijable shella (npr. `$PATH`, `$USER`, `$PWD` ili `$OLDPWD`) koje se mogu pregledati naredbom `env`.
- Parametri tj. argumenti komandne linije označeni su sa `$1`, `$2`, `$3` itd.
  - `$@` je `"$1" "$2" "$3"`
  - `$*` je `"$1 $2 $3"`
  - `$#` je broj parametara

#### Praznine

Znak jednakosti (=) koristi se za pridjeljivanje vrijednosti i za usporedbu.

- Kod pridjeljivanja vrijednosti znak = nije omeđen prazninama:

```
val=33
```

```
str="Neki string"
```

- Kod uspoređivanja, znak = je omeđen prazninama:

```
if [ $str = "Nekistring" ]; then
    echo "je"
else
```

```
        echo "nije"  
fi
```

**Zadatak 9.1** Napišite (i proučite) skriptu naziva `prva` i izvršite je s parametrima: `./prva jen dva`. Što skripta radi?

```
#!/bin/bash  
echo Naziv skripte: $0  
echo Prvi parametar: $1  
echo Drugi parametar: $2  
echo Broj parametara: $#
```

**Zadatak 9.2** Napišite (i proučite) skriptu druga i izvršite je. Što skripta radi?

```
#!/bin/bash  
echo "$USER"  
echo "Sadržaj trenutnog direktorija, $PWD"  
ls -l # list files
```

**Zadatak 9.3** Napišite (i proučite) skriptu treća i izvršite je. Što skripta radi?

```
#!/bin/bash  
X=3  
Y=4  
echo $((X+Y))  
echo $((X-Y))  
echo $((X*Y))  
echo $((X/Y))  
echo $((X%Y))
```

**Zadatak 9.4** Modificirajte skriptu tako da vrši osnovne aritmetičke operacije nad argumentima komandne linije.

## Uvjetno izvršavanje

Uvjetno izvršavanje je izvršavanje naredbe ili skupa naredbi u ovisnosti o povratnoj vrijednosti druge naredbe. Realizira se pomoću `if then else fi` notacije.

```
if test-commands; then  
    consequent-commands;  
[elif more-test-commands; then  
    more-consequents;]  
[else alternate-consequents;]  
fi
```

Uvjetno izvršavanje pomoću uglatih zagrada []. Sintaksa je [ <IZRAZ> ] (uočite razmak nakon otvorene uglate zgrade).

**Primjer 9.1** Izvršite sljedeću skriptu. Što radi?

```
#!/bin/bash
A=101
if [ $A = 10 ]
then
    echo A = 10
else
    echo A != 10
fi
```

**Primjer 9.2**

```
#!/bin/bash
# provjera postoji li datoteka /etc/passwd

if [ -e /etc/passwd ]; then
    echo "Postoji"
else
    echo "Ne postoji"
    exit 1
fi
```

**Zadatak 9.5** Izmijenite prethodni zadatak tako da se parametri zbroje ako je prvi parametar manji od drugog parametra, pomnože ako su jednaki, a oduzmu ako je prvi parametar veći od drugog parametra. Skriptu nazovite script.sh. Napomena: operatori "<" i ">" ne mogu se koristiti izravno jer ih bash tumači kao operatore preusmjerenja. Stoga je u zadatku potrebno koristiti njihove "tekstualne" varijante prikazane u donjoj tablici:

-lt	manje od
-le	manje ili jednako
-gt	veće od
-ge	veće ili jednako

```
# primjer: if [ $X -lt 10 ]; then ...
```



## Vježba 9

### Procesi

#### Proces

Proces je program u izvršavanju zajedno sa svojim virtualnim adresnim prostorom. Svaki proces ima pridružene i sljedeće vrijednosti:

- ID procesa (PID)
- ID roditelja (PPID)
- ID korisnika i grupe
- environment varijable
- listu otvorenih datoteka
- tekući direktorij
- itd.

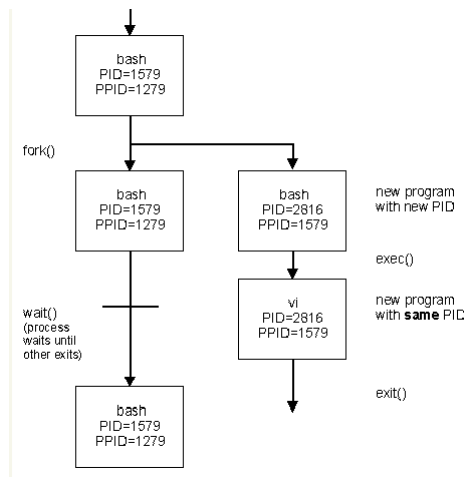
#### Pokretanje procesa

Svaki proces je pokrenut od strane nekog drugog procesa. Jedini izuzetak je proces `init` koji pokreće kernel i ima PID 1. Sama ljuska (engl. shell) je proces i njen PID se može dobiti s: `echo \$$`

Iz ljuske se procesi pokreću izvršavanjem komandne linije. Npr. u komandnoj liniji napišite `bash`. Otvorit će se podljuska (engl. subshell).

**Zadatak 10.1** Izvršite sljedeće akcije:

- Naći PID aktivne ljuske. Izvršiti naredbu `history`.
- Pokrenite novu `bash` podljusku (engl. `subshell`).
- Naći PID novootvorenoog subshella. Izvršiti naredbu `history`. Što zaključujete?
- Izaći iz novootvorenog shella naredbom `exit`. Provjeriti PID shella u kojem se nalazite.



Procesi se mogu izvršavati na dva načina:

- procesi u prednjem planu (eng. `foreground`)
- procesi u pozadini (eng. `background`)

Kada se proces izvršava u prednjem planu, ljuska koja ga je pokrenula treba čekati da proces završi kako bi se mogli pokretati ostali procesi. S druge strane, kada se proces izvršava kao pozadinski, ljuska ne mora čekati završetak procesa (odzivni znak se odmah pojavi). Pozadinski procesi se pokreću tako da se na kraj komandne linije doda znak ampersand (`&`).

Prebacivanje procesa iz prednjeg plana u pozadinu:

1. Suspendirati proces (`Ctrl+z`)
2. Unijeti naredbu `bg`

**Zadatak 10.2** Pokreniti program za uređivanje teksta `xed` ili `gedit` i prebaciti ga u pozadinu.

**Zadatak 10.3** Pokrenuti naredbu `jobs` kojom se ispisuju svi pozadinski ili suspendirani procesi.



## Nadgledanje procesa

- `ps` - ispisuje sve procese (process status). Neke opcije:
  - `a` - svi procesi povezani sa terminalom
  - `x` - svi ostali procesi
  - `u` - detaljniji ispis
  - `-e` - svi procesi
  - `-f` - full format
  - `-l` - kompletan ispis

Najčešći oblik korištenja `ps` naredbe je `ps aux` ili `ps -ef`

- `pstree` - ispisuje procese u obliku stabla
- `pidof` - ispisuje PID procesa
- `top` - ispisuje trenutno pokrenute procese i informacije o memoriji i CPU (izlaz s tipkom `q`).

**Zadatak 10.4** Izlaz naredbe `pstree` spremite u datoteku `stablo.txt`.

**Zadatak 10.5** Pokrenite neki program (npr. `gedit`). Nađite PID i PPID procesa. Ispišite samo one linije koje se odnose na taj proces (koristite naredbe `ps` i `grep`).

**Zadatak 10.6** Izvršite sljedeće zadatke:

- Izlistajte sve procese povezane s terminalom.
- Izlaz sortirajte po stupcu `COMMAND`.
- Naredbom `wc` prebrojite linije u prethodnom ispisu.
- Ponovite sortiranje, s tim da isključite zaglavlje koristeći naredbu `tail`.
- Rezultat ispisa spremite u datoteku.

## Završetak procesa

Procesi završavaju zbog dva razloga:

- proces sam završava, automatski ili zbog korisničke intervencije
- drugi proces šalje signal procesu i tako ga terminira (npr. naredbom `kill`)

Kill signal Najvažniji signali:

---

Signal	Kratica	Značenje	Akcija
02	Ctrl+c	interrupt	end process
09		kill	end process (nepovratno)
15		terminate	end process

---

Sintaksa je `kill -SIGKILL PID`. Predefinirani signal je 15.

**Zadatak 10.7** Ubijte `gedit` iz komandne linije.

# Vježba 10

## Rad na udaljenom računalu

Ova vježba radi se na udaljenom poslužitelju, koristeći program Putty koji komunicira s poslužiteljem na adresi 192.168.4.4 koristeći SSH protokol.

## Spajanje na poslužitelj

Prilikom spajanja na poslužitelj, studenti se spajaju koristeći korisničko ime i lozinku koje dobiju od nastavnika.

## Rad na poslužitelju

Svaki korisnik na poslužitelju ima svoj direktorij (/home/username, odnosno ~) i sve radnje treba obavljati unutar njega. Student ne bi trebao napuštati svoj direktorij, a u niti jednoj situaciji nije dozvoljeno pristupati tuđim podacima. Općenito nije dozvoljeno na bilo koji način ugrožavati sigurnost ili remetiti rad sustava, te pristupati datotekama izvan vlastitog direktorija ili direktorija /materijali.

### Primjer 11.1 Pozicioniranje u home direktorij

- cd ~
- pwd
- ls

- `ls ~`
- `ls /home/102`

**Zadatak 11.1** Unutar direktorija `~` napravite poddirektorij `zad1` i u njemu riješite sljedeće zadatke:

- Ispitajte koji je tekući direktorij.
- Napravite direktorij `~/zad1/tmp`
- Promijenite tekući direktorij u novokreirani direktorij.
- u direktoriju `tmp` napravite datoteke `prva` `druga` `treca`.
- Isprobajte naredbu `tree` i njen izlaz preusmjerite u datoteku `treeoutput`
  
- Kopirajte datoteku `prva` u direktorij `~`
- Preimenujte datoteku `prva` u `prvatmp`
- Pomaknite datoteku `druga` u direktorij `~` i preimenujte je u `druga2` koristeći samo jednu naredbu `mv`.
- Kopirajte direktorij `zad1` i sve datoteke u njemu u direktorij `zad1b`.

**Zadatak 11.2** U direktoriju `~` napravite direktorij `zad2` i uđite u njega.

U direktoriju `zad2` kreirajte datoteku sa nastavkom `.txt` u kojoj su zapisane datoteke iz direktorija `/etc` koje počinju sa slovom `a`.

**Zadatak 11.3** U direktoriju `~` napravite direktorij `zad3` i uđite u njega. Datoteku `/materijali` `/mjesta` kopirajte u tekući direktorij. Datoteci `mjesta` dodijelite takva prava da samo vlasnik može čitati sadržaj datoteke.

**Zadatak 11.4** U direktoriju `~` napravite direktorij `zad4` i uđite u njega. Datoteku `/materijali` `/mjesta` kopirajte u tekući direktorij. Sadržaj datoteke `mjesta` sortirajte po abecedi i izlaz upišite u datoteku `mjesta_sort`

## Prebacivanje povijesti naredbi na sustav Moodle

Naredbom `sharehistory` uploadajte povijest naredbi na web (`sharehistory` je skripta i nije standardna Linux naredba). Otvorite dobivenu poveznicu u pregledniku i kopirajte povijest u tekstualnu datoteku koju ćete predati na Moodle prema uobičajenim pravilima.