1. **Pravljenje, brisanje direktorijuma i izlistavanje njihovog sadržaja. Prvi koraci u stablu direktorijuma.**

Važno je znati da Unix pravi razliku između velikih i malih slova.

U imenima direktorijuma i fajlova se ne preporučuje korišćenje nekih specijalnih znakova: **/**, **[Space]**, **(**, **)**, **[**, **]**, **{**, **}**, **'**, **"**, **+**, **\***, **?**, **<**, **>**, **$**, **\**, **,**. Ako je potrebno imati imena koja bi se sastojala iz više reči, umesto [Space] koristi se **\_**, **.** ili **-**.

Važno je napomenuti da se ne moramo podržavati pravila za imena na koje smo se navikli u DOS-u: <nekoliko karaktera>.<ekstenzija od tri karaktera> Npr. abcdefg.exe. U ime jednog fajla se mogu staviti i više tačaka, tako da npr. abcdefg.exe.ab.c jeste validno ime u Unix-u. Da bi se ljudski korisnici lakše snalazili, ipak se preporučuje pridržavanje nekih pravila oko davanja imena, npr jedna mogućnost je davanje imena kao u DOS-u.

Struktura jedne Unix naredbe izgleda ovako:

<naredba> <opcije> <parametri>

U Unix-u umesto \ (backslash) stoji / (slash) u putanji nekog fajla. Ne postoje C: ili D: drajveri, nego:

/ koreni direktorijum

. aktuelni direktorijum

.. roditeljski direktorijum

~ početni direktorijum (npr. /home/student)

***Naredba clear***

Briše ekran.

***Naredba pwd (print working directory)***

Apsolutna putanja aktuelnog direktorijuma.

1. **Zadatak**

Nakon prijave pogledati u kojem direktorijumu se nalazi korisnik!

Rezultat naredbe pwd bi mogao biti:

[student@drvo1 ~]$ **pwd**

/home/student

Prilikom prijave na sistem kao *student*, korisnik se automatski nalazi u svom početnom direktorijumu.

1. **Zadatak**

Analizirati prompt komandnog interpretera (niz karaktera koji je ispred kurzora)!

[student@drvo1 ~]$

U bash shell-u podrazumevamo sledeći izgled prompta: prva vrednost je korisničko ime (npr. student), zatim posle znaka **@** sledi ime računara u mreži (npr. drvo1). Dalje, tu je ime aktivnog direktorijuma (ne cela putanja samo ime direktorijuma; u ovom slučaju je to početni direktorijum, tako da se vidi znak **~**), na kraju je oznaka za dozvole i prava prijavljenog korisnika: **$** - običan korisnik, **#** - superkorisnik, administrator odnosno root.

***Naredbe mkdir, rmdir, ls i cd***

mkdir naziv napravi novi direktorijum (make directory)

mkdir naziv1 naziv2 napravi više novih direktorijuma u jednom koraku

rmdir naziv brisanje direktorijuma (remove directory)

ls izlistavanje sadržaja direktorijuma (list)

cd menjanje aktuelnog direktorijuma (change directory)

***Napomena:*** *u primerima imena direktorijuma se zadaju pomoću njihovih relativnih putanja. To znači da su imena validna u odnosu na aktuelni direktorijum. Npr. korisnik se nalazi u direktorijumu* ***/home/student/*** *i želi preći u folder* ***/home/student/TORBA****, putanja se jednostavno zadaje kao* ***TORBA****. U slučaju apsolutne putanje direktorijum se zadaje počev od koren direktorijuma root, npr.* ***/home/student/TORBA****.*

1. **Zadatak**

Kreirati direktorijum TORBA i proveriti da li je stvarno nastao!

[student@drvo1 ~]$ **mkdir TORBA**

[student@drvo1 ~]$ **ls**

TORBA

1. **Zadatak**

Obrisati direktorijum TORBA i proveriti da li je stvarno nestao!

[student@drvo1 ~]$ **rmdir TORBA**

[student@drvo1 ~]$ **ls**

1. **Zadatak**

U jednom koraku kreirati dva direktorijuma, proveriti rezultat naredbe, zatim obrisati ih!

[student@drvo1 ~]$ **mkdir SVESKA TORBA**

[student@drvo1 ~]$ **ls**

SVESKA TORBA

[student@drvo1 ~]$ **rmdir TORBA**

[student@drvo1 ~]$ **rmdir SVESKA**

[student@drvo1 ~]$ **ls**

1. **Zadatak**

Ponovo kreirati direktorijume TORBA i SVESKA, a u istom koraku u direktorijumu TORBA napraviti direktorijum DZEP, u direktorijumu SVESKA folder STRANICA! Proveriti rezultat!

[student@drvo1 ~]$ **mkdir TORBA SVESKA TORBA/DZEP SVESKA/STRANICA**

[student@drvo1 ~]$ **ls**

SVESKA TORBA

[student@drvo1 ~]$ **cd TORBA/**

[student@drvo1 TORBA]$ **ls**

DZEP

[student@drvo1 TORBA]$ **cd ../SVESKA/**

[student@drvo1 SVESKA]$ **ls**

STRANICA

1. **Zadatak**

Preći u direktorijum STRANICA, proveriti da li se tamo nešto nalazi i sa tog mesta izlistati sadržaj direktorijuma TORBA!

[student@drvo1 SVESKA]$ **cd STRANICA**

[student@drvo1 STRANICA]$ **ls**

[student@drvo1 STRANICA]$ **ls ../../TORBA/**

DZEP

1. **Zadatak**

Obrisati svaki napravljeni folder!

[student@drvo1 STRANICA]$ **cd ..**

[student@drvo1 SVESKA]$ **rmdir STRANICA**

[student@drvo1 SVESKA]$ **cd ..**

[student@drvo1 ~]$ **rmdir SVESKA**

[student@drvo1 ~]$ **rmdir TORBA/DZEP**

[student@drvo1 ~]$ **rmdir TORBA**

[student@drvo1 ~]$ **ls**

***Magični taster Tab***

1. **Zadatak**

Upoznajte se sa efikasnim korišćenjem tastera Tab u komandnom interpreteru bash. Pomoću ovoga ući u direktorijum /home/student!

Pritiskom na Tab shell pokuša da završi upisani početak imena fajla. Ako je to jednoznačno (postoji samo jedna mogućnost) upisaće ime a ako nije, izdaje jedan zvučni signal bip i nakon ponovnog pritiska na Tab dobije se lista mogućih imena fajlova. Upotpunjavanje imena fajlova je donekle inteligentno: na početku redova (prva reč) traži samo u izvršnim fajlovima i naredbama odgovarajuće ime, (pregledaće PATH), a kad je reč o argumentima gleda svaki fajl aktuelnog (ili odgovarajućeg započetog) direktorijuma.

Ako se na početku reda pritisne taster Tab dva puta, sistem će pitati da li korisnik stvarno želi da se ispišu svih 2167 naredbi (naravno ovaj broj se menja u odnosu na distribuciju operativnog sistema).

[student@drvo1 ~]$

Display all 2167 possibilities? (y or n)

Promena direktorijuma se postiže naredbom *cd*. Treba da se upiše:

[student@drvo1 ~]$ **cd /h<tab><tab>**

Pošto u direktorijumu root postoji samo jedan folder čije ime počinje sa slovom *h* (to je folder home), shell upotpunjuje početak i stavlja znak / dajući do zanja da je upotpunjavanje jednoznačno. Ako se u ovoj situaciji dva puta pritisne taster Tab, shell ispisuje ceo sadržaj direktorijuma /home/ npr.:

[student@drvo1 ~]$ **cd /home/<tab><tab>**

ob/ laboratory/ output student/

ftp-docs/ lost+found/ proba

Samo jedno ime fajla počinje na slovo s, nastavlja se unos slovom s i pritiskom na Tab:

[student@drvo1 ~]$ **cd /home/student/**

Komandni interpreter upotpunjuje upis na direktorijum student. Naredba se aktivira pritiskom na taster Enter.

***Fajlovi: pravljenje tekstualnog fajla, dodavanje teksta na kraj fajla, kopiranje, premeštanje i brisanje fajlova***

U Unix/Linux-u je sve fajl: fajl je i direktorijum. To je jedan specijalni fajl u kome Unix/Linux drži administrativne informacije o ostalim fajlovima. Korisnik ne može da menja njegov sadržaj po volji: sistem Unix/Linux-a to vrši paralelno sa operacijama sa fajlovima, tako da sadržaj direktorijuma u svakom trenutku odgovara stvarnoj strukturi fajlova.

U Unix/Linux-u je sve fajl, ali nije svaki fajl iste prirode. Fajlovi se mogu svrstati u nekoliko tipova:

- običan fajl

d direktorijum

l simbolički lanac (link)

***Naredba cat***

cat naziv ispisivanje sadržaja fajla na ekran (engl. catenate)

cat >naziv pravljenje tekstualnog fajla (> je preusmeravanje izlaza)

cat >>naziv dodavanje teksta na kraj fajla

cat naziv1 >naziv2 kopiranje sadržaja fajla *naziv1* u fajl *naziv2* (ako fajl *naziv2* već postoji, gubi se stari sadržaj)

cat naziv1 >>naziv2 sadržaj fajla *naziv1* se dopisuje na kraj fajla *naziv2*

cat naziv1 naziv2 ispisivanje fajlova na ekran u datom redosledu

1. **Zadatak**

Kreirati tekstualni fajl sa sadržajem:

Olovka je crvena

i dugacka 10cm.

[student@drvo1 ~]$ **cat >olovka**

Olovka je crvena

i dugacka 10cm.

^D

Kombinacija tastera Ctrl+D (skraćeno ^D) dodaje karakter EOF na kraj fajla. Ovaj karakter zatvara fajl i vraća prompt.

[student@drvo1 ~]$ **ls**

[student@drvo1 ~]$ **cat olovka**

1. **Zadatak**

Dodati novi red u već napravljeni fajlu *olovka*!

[student@drvo1 ~]$ **cat >>olovka**

U torbi je jos i gumica.

Proverava se sadržaj fajla *olovka*:

[student@drvo1 ~]$ **cat olovka**

1. **Zadatak**

Prekopirati fajl *olovka* u direktorijum *SVESKA* pod imenom *olovka2*! Preći u taj direktorijum i proveriti da li je operacija bila uspešna!

Prvo se pravi direktorijum *SVESKA*:

[student@drvo1 ~]$ **mkdir SVESKA**

Izlistava se sadržaj fajla *olovka* i rezultat se upisuje u fajl *olovka2* koji se nalazi u folderu *SVESKA*.

[student@drvo1 ~]$ **cat olovka > SVESKA/olovka2**

Prelazi se u direktorijum *SVESKA* i proverava se sadržaj foldera:

[student@drvo1 ~]$ **cd SVESKA**

[student@drvo1 SVESKA]$ **ls**

[student@drvo1 SVESKA]$ **cat olovka2**

1. **Zadatak**

Sadržaj fajla *olovka2* prepisati u novi sadržaj:

U svesci ne bi trebalo da bude i olovaka...

[student@drvo1 SVESKA]$ **cat >olovka2**

U svesci ne bi trebalo da bude i olovaka...

^D

[student@drvo1 SVESKA]$ **cat olovka2**

1. **Zadatak**

Dopisati sadržaj fajla *olovka2* fajlu *olovka* koji se nalazi u početnom direktorijumu!

[student@drvo1 SVESKA]$ **cat olovka2 >>../olovka**

[student@drvo1 ~]$ **cd ..**

[student@drvo1 ~]$ **ls**

[student@drvo1 ~]$ **cat olovka**

***Naredbe cp i mv***

cp kopiranje fajlova (copy)

mv premeštanje fajlova (move)

cp/mv <opcije> <izvor> <cilj>

cp/mv <opcije> <izvori> <ciljni direktorijum>

Ako za cilj zadajemo ime fajla koji već postoji, stari sadržaj fajla se gubi i pod starim imenom će se naći novi fajl.

1. **Zadatak**

Obrisati sadržaj ekrana! Kopirati fajl *olovka* u direktorijum *SVESKA*!

Sadržaj ekrana se briše pomoću naredbe **clear**:

[student@drvo1 ~]$ **clear**

Ako je korisnik u početnom direktorijumu gde je folder SVESKA i fajl *olovka*, može se izdati sledeća naredba:

[student@drvo1 ~]$ **cp olovka SVESKA/olovka**

[student@drvo1 ~]$ **cd SVESKA**

[student@drvo1 SVESKA]$ **ls**

[student@drvo1 SVESKA]$ **cat olovka**

1. **Zadatak**

U početnom direktorijumu kreirati folder *TORBA*, zatim iz direktorijuma *SVESKA* prebaciti oba fajla u folder *TORBA*!

Prvo treba preći u početni direktorijum, zatim se pravi folder *TORBA*:

[student@drvo1 ~]$ **cd /home/student**

[student@drvo1 ~]$ **mkdir TORBA**

[student@drvo1 ~]$ **ls TORBA**

U novom folderu ne postoji ništa. Prebacuju se dva fajla iz direktorijuma *SVESKA*:

[student@drvo1 ~]$ **mv SVESKA/olovka SVESKA/olovka2 TORBA**

[student@drvo1 ~]$ **cd TORBA**

[student@drvo1 TORBA]$ **ls**

[student@drvo1 TORBA]$ **cat olovka**

1. **Zadatak**

Preimenovati fajl *olovka2* u *svasta*!

[student@drvo1 TORBA]$ **mv olovka2 svasta**

[student@drvo1 TORBA]$ **ls**

***Naredba rm***

1. **Zadatak**

Obrisati sve napravljene fajlove i direktorijume!

[student@drvo1 ~]$ **rm olovka**

[student@drvo1 ~]$ **rm svasta**

[student@drvo1 ~]$ **cd ..**

[student@drvo1 ~]$ **rmdir TORBA**

[student@drvo1 ~]$ **rmdir SVESKA**

[student@drvo1 ~]$ **rm olovka**

[student@drvo1 ~]$ **ls**

***Najvažnija naredba: man***

Sistem za pomoć u Unix/Linux-u daju stranice priručnika (manuala). Naredbom *man* se može učitati odgovarajuća stranica za pomoć. O svim naredbama, pomoćnim programima i funkcijama pomoć se traži na sledeći način:

man <naredba>

I o samoj naredbi man se može doći do informacija: man man

1. **Zadatak**

Pogledati stranice priručnika za naredbu *echo*! Analizirati delove priručnika!

[student@drvo1 ~]$ **man echo**

Opisi se dele u 8 poglavlja:

1. Obični programi

2. Pozivi sistema

3. Bibliotečke funkcije na nivou korisnika

4. Specijalni fajlovi

5. Formati fajlova

6. Igrice

7. Razni opisi

8. Informacije potrebne za održavanje sistema

Mogu postojati opisi sa istim nazivom u više poglavlja. Npr. može postojati program i funkcija sistema sa istim imenima. U takvim slučajevima treba da se zadaje i broj poglavlja:

man <poglavlje> <naredba>

Isprobati naredbe *man printf* i *man 3 printf*!

Za izlaz iz listinga naredbe man može se pritisnuti taster *q* (q kao skraćenica za englesku reč *quit*).

1. **Zadatak**

Potražiti u kom priručniku se može pronaći ključna reč *reset*!

Ključne reči se traže pomoću opcije -k (k kao skraćenica za englesku reč *keyword* – ključna reč):

[root@drvo1 ~]$ **man -k reset**

clearerr [ferror] (3) - check and reset stream status

feof [ferror] (3) - check and reset stream status

***Zaustavljanje i ponovo pokretanje sistema (rebooting)***

1. **Zadatak**

Nakon vežbe zaustaviti sistem!

U grafičkom okruženju zaustavljanje sistema se može izvršiti na sledeći način (slika 24):

K Menu  Log Out  Turn Off Computer

U karakterskom režimu rada ili u x-terminalu grafičkog okruženja sistem se može zaustaviti naredbom *shutdown* i odgovarajućim opcijama.

[student@drvo1 ~]$ **shutdown –h now**

Opcijom **–h** postiže se da nakon zaustavljanja sistema isključi napajanje računara (turn off). Opcija **now** određuje vreme nakon kojeg naredba *shutdown* pokušava zaustaviti sistem. Now na engleskom značisada, tako da odmah nakon izdavanja naredbe sistem treba da zaustavi programe koji se trenutno izvršavaju i da isključi računar.

*Slika 24*

Naredba *shutdown –h now* za zaustavljanje sistema ustvari poziva nulti nivo izvršavanja. Ovo se može i uneti sa tastature i dobija se isti rezultat:

[student@drvo1 ~]$ **init 0**

1. **Zadatak**

Kako se sistem može ponovo pokrenuti?

Ponovno pokretanje sistema je ustvari izdavanje naredbi redom za zaustavljanje, isključivanje i pokretanje sistema. Jednostavniji način je "zamoliti" *shutdown* da nakon zaustavljanja ponovo pokrene sistem. To se postiže opcijom **-r**: *shutdown -r now*. Najveći broj sistema Linux izvršava naredbu *shutdown -r now* kada se na konzoli pritisne kombinacija tastera Ctrl+Alt+Del odnosno i u ovom slučaju se sistem ponovo pokreće. Može da se odredi šta da se desi kao odgovor na kombinaciju tastera Ctrl+Alt+Del, npr. u slučaju višekorisničkog sistema svrsishodno je osigurati i malo čekanja pre izvršavanja naredbe. Na sistemima čiji konzola stoji svakome na raspolaganju može se podesiti i da se ne desi ništa pritiskom na tastere Ctrl+Alt+Del.

Za ponovno pokretanje sistema izdaje se naredba:

[student@drvo1 ~]$ **shutdown –r now**

U grafičkom okruženju ponovno pokretanje sistema može da se izvrši na sledeći način:

**K Menu**  **Log Out**  **Restart Computer**