

SKYRIUS 4 UNIX Failų Sistema

4.1 -	Trys (pagrindiniai) Unix failų tipai	4-2
4.2 -	Tekstiniai ir dvejetainiai failai, bitai ir baitai	4-4
4.3 -	Medžio pavidalo failų sistema	4-5
4.4 -	BSD standartiniai katalogai ir jų turinys	4-7
4.5 -	System V Unix'o Standartinių katalogų paskirtis	4-10
4.6 -	Katalogo /dev specialių failų naudojimas: tty	4-14
4.7	Namų katalogai	4-15

4.1 Trys (pagrindiniai) Unix failų tipai

Kas yra failas?

Failo supratimas Unix'e yra žymiai platesnis: failas yra bet kuris šaltinis iš kurio duomenys gali būti skaitomi, arba bet kuris "gavėjas", galintis priimti duomenis. Tokiu būdu, terminas *failas* nurodo ne tik duomenų saugyklą diskinio failo pavidale, bet kartu reiškia ir bet kurį fizinį įtaisą. Failas yra ir klaviatūra (įvedimo šaltinis), ir displėjus (atvaizdavimo įtaisas), ir kiekvienas spausdintuvas.

Unix'e yra trys pagrindiniai failų tipai: paprastieji failai, katalogai ir specialūs failai.

PAPRASTIEJI failai yra tai, ką mes suprantame failu siaurąja prasme. Paprastąjį failą sudaro diske (rečiau juostoje) saugomi duomenys. Tai failai, su kuriais dažniausiai tenka dirbti. Pvz., kai jūs teksto redaktoriaus tvarkote dokumentą, tai jūs naudojate du paprastuosius failus: dokumento ir redaktoriaus programos.

Kitas failo tipas – KATALOGAS. Katalogas tai failas, kuriame saugoma kitų failų pasiekimui reikalinga informacija. Tik conceptualiai į katalogą įeina kiti failai. Vartotojas nekuria katalogų ir netalpina juose failų. Jis tik naudoja komandas, kurios paveda tuos darbus Unix'ui. Jeigu jūs pažiūrėsite į katalogo 'vidų', tai pamatysite 'miglotą' informaciją. Geriausias būdas suprasti katalogo sąvoką - įsivaizduoti, kad tai yra rodyklių į kitus failus visuma. Katalogas gali turėti rodykles į kitus katalogus. Tai leidžia organizuoti failus į hierarchinę sistemą. Kaip matysime vėliau, Unix failų sistema yra organizuota kaip didžiulė failų ir katalogų hierarchija.

Paskutinis failų tipas - tai SPECIALŪS, arba PRIETAISŲ failai, kurie yra vidinė fizinių prietaisų reprezentacija. Pvz., klaviatūra, displėjus, spausdintuvas, disko tvarkyklė - visi jūsų sistemos prietaisai gali būti pasiekiami kaip failai. Pasiusti duomenis į ekraną, ar į kitą terminalą, galima paprasčiausiai juos rašant į failą, kuris reprezentuoja atitinkamą prietaisą.

Pvz., `ls > /dev/pts/u`

Unix'e yra tokie failų tipai:

Tipas	Žymėjimas
Regulai	(-)
Directory	(d)
Hard Links	(-)
Soft Links	(l)
Block special	(b)
Character special	(c)
Socket	(s)
Named pipe	(p)

Kiekvienas sistemos failas turi taip vadinamą inode bloką, kuriame yra informacija apie failą. Inode blokas turi tokią struktūrą: ----- —

```

struct stat {
    dev_t      st_dev;          /* device inode resides on */
    ino_t      st_ino;          /* this inode's number */
    u_short    st_mode;         /* protection */
    short      st_nlink;        /* number of hard links to the file */
    short      st_uid;          /* user-id of owner */
    short      st_gid;          /* group-id of owner */
    dev_t      st_rdev;         /* the device type, for inode that is
    device */
    off_t      st_size;         /* total size of file */
    time_t     st_atime;        /* file last access time */
    int        st_spare1;
    time_t     st_mtime;        /* file last modify time */
    int        st_spare2;
    time_t     st_ctime;        /* file last status change time */
    int        st_spare3;
    long       st_blksize;      /* optimal blocksize for file system i/o ops */
    long       st_blocks;       /* actual number of blocks allocated */
    long       st_spare4;
    u_long     st_gennum;       /* file generation number */
};

```

Pagrindiniai šios struktūros laukai `st_mode` (the permission bits), `st_uid` - UID, `st_gid` - GID, ir `st_*time` (assorted time fields).

Failo laikai

Unix įrašo į *inode* tris failo laikus, kurie vadinami *ctime*, *mtime*, ir *atime*.

ctime rodo paskutinį *inode* bloko keitimo laiką;
mtime rodo paskutinį failo modifikavimo laiką;
atime rodo paskutinį kreipimosi į failą laiką.

Inod'o *ctime* kinta visuomet, kai į failą yra rašoma, arba kinta jo apsaugos ar nuosavybės bitai. Paprastai *ctime* geriau pažymi failo modifikavimą, negu lauką *mtime*.

Failo laikai yra svarbūs ir įvairiais būdais naudojami sistemos administravimo metu. Pvz., darant diskų kopijas.

4.2 Tekstiniai ir dvejetainiai failai, bitai ir baitai

Paprasti failai gali būti labai įvairūs, kadangi labai įvairūs gali būti juose saugomi duomenys. Tačiau, plačiąja prasme, paprastus failus galima skirti į dvi grupes: tekstinius failus ir dvejetainius failus.

TEKSTINIAI failai yra paprasti failai, kuriuose saugomi tik ASCII simboliai: duomenys, kuriuos galima surinkti klaviatūros pagalba. Trumpai tariant, skirtingus ASCII simbolius reprezentuoja 128 skirtingi kodai: didžiosios ir mažosios raidės, skaičiai, tarpo simbolis, tabuliacijos simbolis, skyrikliai, o taip pat valdymo simboliai. Praktinė taisyklė: jei paprastame faile saugomi duomenys, kuriuos jūs galite suprasti ir tvarkyti teksto redaktoriaus pagalba, tai tas failas yra tekstinis. Priešingu atveju yra dvejetainis failas.

Tekstinio failo kiekvienas ASCII kodas saugomas viename baite. Tokiu būdu failas, turintis 1000 simbolių, yra 1000-ties baitų ilgio. Griežtai kalbant, ASCII kodas naudoja septynis bitus vienam simboliui (baitė yra 8 bitai). Kiekvieno baito aštuntas bitas (kairysis) visuomet turi nulinę reikšmę. Apibendrinant galima pasakyti, kad tekstinis failas saugo ASCII simbolius ir naudoja kiekvienam simboliui septynis baito bitus. Dvejetainis failas saugo duomenis, kurių kodams naudojami visi aštuoni kiekvieno baito bitai.

Katalogai ir Pakatalogiai

Failų organizavimui į hierarchinę sistemą naudojami katalogai. Mes grupuojame failus į grupes ir saugome kiekvieną grupę atskirame kataloge. Kadangi katalogas taip pat yra failas, tai kataloge gali būti saugomi kiti katalogai.

4.3 Medžio pavidalo failų sistema

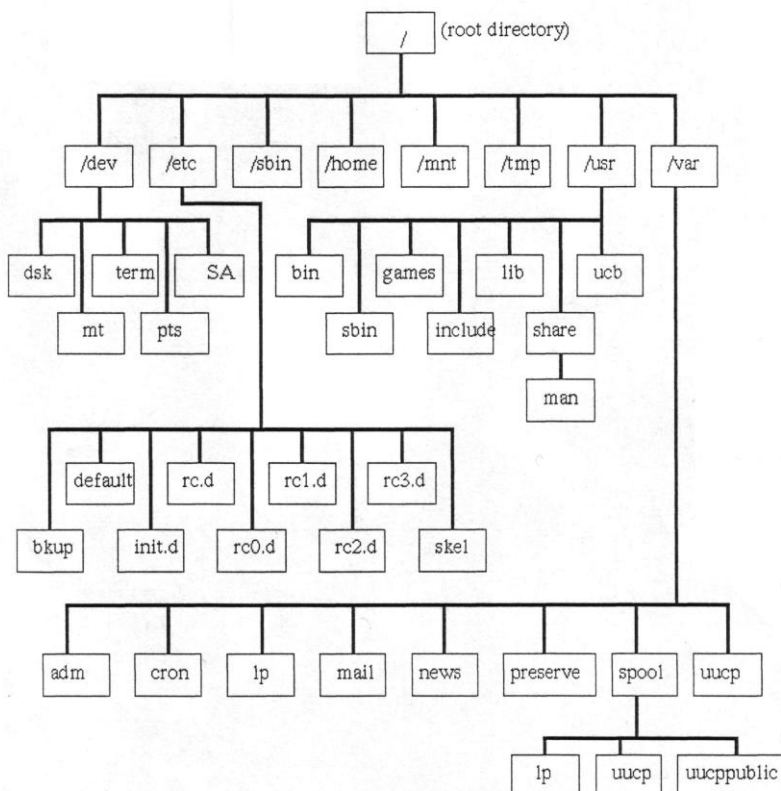
Unix sistemą sudaro daugybė failų, kurie organizuojami į tam tikrą struktūrą, naudojant katalogus ir pakatalogius. Unix failų sistema pagrįsta vienu pagrindiniu katalogu, vadinamu šakniniu katalogu (ROOT DIRECTORY). Šakninis katalogas yra protėvis visų kitų sistemos katalogų.

Pateiktoje schemeje parodytas Unix failų sistemos turinys. Tokius katalogus naudoja Berkeley Unix pagrindu sukurtos sistemos. System V, arba kitu pagrindu sudarytos sistemos gali nežymiai skirtis nuo čia pateiktos. Tai ką jūs matote yra tik nuogas skeletas. Visos Unix sistemos turės ir kitus, sistemų administratorių sukurtus katalogus.

4.3 Medžio pavidalo failų sistema

Unix sistemą sudaro daugybė failų, kurie organizuojami į tam tikrą struktūrą, naudojant katalogus ir pakatalogius. Unix failų sistema remiasi vienu pagrindiniu katalogu, vadinamu ROOT DIRECTORY (šakninis katalogas). Šakninis katalogas yra protėvis visų kitų sistemos katalogų.

Pateiktoje schemeje parodytas Unix failų sistemos turinys. Tokius katalogus naudoja Berkeley Unix pagrindu sukurtos sistemos. System V, arba kitu pagrindu sudarytos sistemos gali nežymiai skirtis nuo čia pateiktos. Tai ką jūs matote yra tik nuogas skeletas. Visos Unix sistemos turės ir kitus, sistemų menedžerių sukurtus katalogus.



Nesunku pastebėti, kad Unix failų sistema panaši į apverstą medį. Žodis *root* buvo parinktas tam, kad pabrėžti pagrindinę medžio liemens dalį. Kadangi šakninio katalogo vaidmuo yra labai svarbus, tai šis vardas dažnai turi būti naudojamas komandose. Todėl buvo įvestas žodžio *root* sutrumpinimas / (slash).

Pvz.,

`ls`

komanda parodys visus specifikuoto katalogo failų vardus. Visus šakninio katalogo failus parodo komanda

`ls /`

Šaknyje esantis failas, arba katalogas nurodomi rašant / prieš atitinkamą vardą. Pav., formalus `etc` katalogo vardas yra `/etc`. Katalogas, arba failas, esantis kitame kataloge, nurodomi atskiriant vardus simboliu /. Pav., kataloge `/etc` yra `passwd` failas. Formalus šio failo vardas yra `/etc/passwd`. Panašiai, jeigu `/usr` kataloge yra kitas katalogas `dict`, o jame yra failas `words`, tai formalus šio failo vardas yra `/usr/dict/words`.

Simbolis / turi dvi skirtingas reikšmes. Failo vardo pradžioje jis reiškia šakninį katalogą. Failo vardo viduryje jis atlieka skyriklio vaidmenį.

Gera supratus pateiktą Unix katalogų medį, galima pastebėti, kad, visi katalogai, išskyrus šakninį (/), priklauso kitam katalogui. Tokiu būdu, visi katalogai, išskyrus šakninį, yra pakatalogiai. Paprastumo dėlei mes juos vadiname katalogais.

Root kataloge yra trys, `etc`, `tmp`, ir `usr` vardais pavadinti pakatalogiai. Tokie vardai buvo parinkti todėl, kad jie yra trumpi ir lengvai surenkami klaviatūroje. Tačiau jie sunkiai ištariami. Todėl susitarta `etc` vadinti *etcetera*, `tmp` - *temp*, o `usr` - *user*. Pav., failo `/etc/passwd` vardas tariamas *slash etcetera slash password*.

Root katalogas

Tai visos sistemos pagrindas. Paprastai jame talpinami tik kiti katalogai. Tačiau šaknyje būna vienas svarbus failas: programa, kuri vadinama Unix branduoliu (kernel). Berkeley Unix sistemose ši programa vadinama **vmunix** (virtual memory Unix). Unix sistemos menedžeris prisijungia prie sistemos **root** vardu. Dabar yra aiški šito vardo kilmė: *root* katalogas yra aukščiausiam failų sistemos lygyje.

4.4 BSD standartiniai katalogai ir jų turinys

Kelias	Turinys
/	'root' katalogas
/bin arba /sbin	minimaliam sistemos funkcionavimui reikalingos komandos
/dev	terminalus, diskus, modemus, etc. reprezentuojantys įtaisų failai
/etc	sistemos startavimo ir konfigūracijos failai
/lib	C kompiliatoriaus bibliotekos
/tmp	laikini failai, kurie išnyksta perkrovus sistemą
/sys	branduolio generavimo darbinė sritis, konfigūracijos failai (BSD)
/proc	visų vykdomų procesų atvaizdai (naujesnėse sistemose)
/stand	Stand-alone utilitos, diskų formataavimo priemonės, etc.
/usr/bin	vykdomieji failai
/usr/games	žaidimai ir kt. (daugelis jų nelabai įdomūs)
/usr/include	C programų h-failai
/usr/5bin	System V komandų analogai skirti BSD sistemoms
/usr/etc	sistemos palaikymo komandos
/usr/sbin	papildomos sistemos palaikymo komandos
/usr/lib	standartinių UNIX programų palaikymo failai
/usr/man	'On-line manual' puslapiai
/var/adm	apskaitos failai, resusų naudojimo įrašai
/var/spool	printerių 'spooling' katalogai, UUCP, mail, etc.
/var/tmp	laikina sritis (failai neišnyksta po sistemos perkrovimo)
/usr/usb	Berkeley utilitos ir programos
/usr/local	lokalį programinę įrangą (jūsų instaliuoti paketai)
/usr/local/adm	lokalūs apskaitos ir šabloniniai failai
/usr/local/bin	lokalūs vykdomieji failai
/usr/local/etc	lokalūs sistemos konfigūravimo failai ir komandos
/usr/local/lib	lokalūs pagalbiniai failai
/usr/local/sbin	lokalios sistemos palaikymo komandos
/usr/local/src	/usr/local/* išeities kodas
/kernel	failai, kurie reikalingi branduolio užkrovimui (Solaris)

/bin kataloge saugomos bazinės programos, iš kurių susideda pati Unix sistema. Čia galima rasti daugelį Unix komandų, t.y. failų, kurie vykdomi pagal atitinkamą komandą. Vardas *bin* gali būti traktuojamas dvejopai: daugelis failų yra dvejetainiai (binary); tai programų saugykla;

/dev kataloge yra specialūs failai, kurie reprezentuoja fizinius įtaisus. Paprastai vartotojai nenaudoja šių failų tiesiogiai. Tačiau norint taip daryti - Unix sistema yra atvira (vėliau bus pateiktas pavyzdys).

/etc katalogas skirtas sistemos administratoriui. Čia saugomi failai ir programos, reikalingi sistemos valdymui. Plačiausiai žinomas yra sistemos slaptažodžių failas *passwd*. Kitas žinomas failas yra *temcap* (kitose sistemose *tenninfo*), t.y. visų terminalų, kuriuos gali naudoti Unix, techninių aprašymų duomenų bazė.

/lib kataloge yra programuotojų naudojamų programų biblioteka.

/lost+found katalogą naudoja speciali programa, kuri tikrina Unix failų sistemą. Kai ši programa randa failą, kuris nepriklauso jokiame kataloge (avarinis atvejis), tas failas talpinamas į *lost+found* katalogą. Tokiu būdu sistemos administratorius gali priimti tolimesnius sprendimus.

/sys kataloge yra taip vadinami sistemos išeities failai. Jie gali dominti sistemos administratorių ir programuotojus.

/tmp katalogas - tai laikina saugykla. Kiekvienas gali laikinai talpinti čia savo failus, tačiau laikas-nuo-laiko visi failai šalinami automatiškai. Paprastai čia talpinami laikini failai, kurie reikalingi programos vykdymui. Pav., naudojant *vi* redaktorių, redaguojamo failo kopija laikoma *l tmp* kataloge.

/usr kataloge yra visa eilė svarbių pakatalogių. Tai vienas iš svarbiausių šaknies pakatalogių. Jis pats suskirstytas į visą eilę pakatalogių.

/usr/bin katalogas, kaip ir jo bendravardis šaknyje, skirta vykdomųjų failų laikymui.

/usr/dict - tai Unix žodyno naudojami failai. Faktiškai - tai žodžių sąrašas.

/usr/games - tai žaidimų ir kitų atrakcijų katalogas.

/usr/include katalogas saugo taip vadinamus C programavimo kalbos 'include' (arba ^h) failus, kuriuos naudoja programuotojai įvairiose programose kompiliavimo metu.

/usr/lib - tai analogiška */lib* biblioteka, svarbi tik programuotojams.

/usr/local - tai sistemos administratoriaus patogumui skirtas katalogas, kuriame gali būti saugomos vietinės reikšmės programos ir dokumentacija.

/usr/man katalogas turi visą eilę pakatalogių ir failų, kuriuos naudoja "Unix on-line manual", t.y. operatyvi ^hhelp' tipo dokumentacijos sistema.

/usr/spool - pakeletės stotelės vaidmenį atliekantis katalogas. Čia laikomi duomenys,

kurie buvo kur nors pasiusti. Pav., kai jūs spausdinate failą, tai tas failas, kartu su sistetine informacija laikomas šiame kataloge. Failas laikomas čia iki tol, kol galės būti atspausdintas. Lygiai taip pat čia patenka ir pasiusti elektroniniu paštu laiškai, kol jie iš tikrųjų 'palieka' kompiuterį.

/usr/src kataloge saugomas jūsų Unix sistemos išeities kodas. Jeigu jūsų Unix yra licenzijuotas (galima pirkti pačią Unix sistemą, arba jos licenziją), tai čia saugomas išeities kodas, leidžiantis "žvilgtelėti į sistemos vidų", daryti pakeitimus ir perkompiliuoti sistemą.

/usr/ucb - čia laikomos programos, kurios buvo sukurtos Kompiuterių mokslo departamente, University of California at Berkeley.

4.5 System V Unix'o Standartinių katalogų paskirtis

/dev

The device directory contains all files that represent peripheral devices. Subdirectories of /dev hold the special files for the access modes of a given type of device.

/dev/dsk

The disk directory holds the device files for disks.

/dev/mt

This directory contains the device files for tape drives.

/dev/term

This directory holds the device files for terminais.

/dev/pts

This directory contains the device files for pseudo-terminals. A pseudo-terminal mimics a terminal although rather than being connected to an actual terminal it is connected to a process. Pseudo-terminals are used by networking programs such as rlogin.

/dev/SA

This directory is now obsolete.

/etc

The etc directory holds system configuration files as well as some administrative files.

/etc/bkup

This directory is used to store dump dates.

/etc/default

The default directory holds files containing the default values for various commands.

/etc/init.d

This directory holds shell scripts that are used to transition between run levels in the system. These scripts are run when the system is booted and brought into multi-user mode.

/etc/rc.d

This directory contains all the files and commands associated with all the run levels for the system. rc0.d, rc1.d, rc2.d, and rcS.d all link back to this file for information.

/etc/rc0.d

Files in this directory are associated with run level 0, which is actually a powered down state.

/etc/rc1.d

This directory holds files that are executed to bring the system to run level 1, or single-user mode. This run level is also referred to as "s" or "S".

/etc/rc2.d

This directory contains files used to transition the system to run level 2. Level 2 is also known as multiuser mode and is the normal mode for the system.

/etc/rc3.d

This directory contains files used to move the system to run level 3.

/etc/skel

This directory contains Standard initialization files for the system.

/sbin

The sbin directory contains system administration files that are needed during the boot process.

/home

Users home directories are typically subdirectories of the /home directory.

/mnt

Mount directory. An empty directory conventionally designed for temporarily mounting disk partitions.

/tmp

Temporary directory, available to all users as a scratch directory. The system administrator should see that all the files in this directory are deleted occasionally. Normally, one of the UNIX startup scripts will clear /tmp.

/usr

This directory typically contains files that are used by the system, as well as subdirectories containing files that need to be shared like man pages and user commands.

/usr/bin

Typically holds Standard Unix utility programs. Under BSD this directory is called /bin.

/usr/sbin

Like /sbin this directory contains Utilities for system administration. The Utilities found in /usr/sbin are usually needed after the system is up rather than at boot time.

/usr/games

This directory contains the Standard UNIX games collection. Some sites choose to remove this directory.

/usr/include

Include files. This directory contains C-language header files which define the C programmer's interface to Standard system features and program libraries. For example, it contains the file stdio.h, which defines the user's interface to the C Standard I/O library. The directory /usr/include/sys contains operating system include files.

/usr/lib

Library directory, for public library files. Among other things, this directory contains the Standard C libraries for mathematics and I/O. This directory may also contain certain configuration files for UNIX services such as mail. Finally, subdirectories of `/usr/lib` contain command and configuration files for various Standard UNIX facilities like `uucp` and `lex` and optional software products.

`/usr/share`

The share directory contains files that are meant to be shared between systems.

`/usr/share/man`

Manual pages directory. This directory contains the online version of the UNIX reference manuals. It is divided into subdirectories for each section of the manual. Under BSD and some System V implementations, `/usr/man` contains several man subdirectories holding the raw source for the manual pages in that section and the cat subdirectories storing the processed versions. The latter can be cleared to save space; they will be filled only as manual pages are actually accessed. If present, the file `/usr/man/whatis` contains a database used by the `whatis` and `apropos` (`man -k`) commands.

`/usr/ucb`

A directory that contains Standard UNIX commands originally developed at the University of California, Berkeley. This directory sometimes includes subdirectories for separate file types (bin for binaries, lib for libraries, and so on).

`/var`

This directory holds spooling directories that are kept in `/usr/spool` under BSD.

`/var/adm`

Administrative directory (home directory of the special `adm` user). This directory contains the UNIX accounting files and various system logging files.

`/var/cron`

This directory contains files relevant to the cron daemon.

4.6 Katalogo /dev specialių failų naudojimas: tty

Visi specialūs failai saugomi /dev kataloge. Pamatyti šių failų vardus leidžia komanda

Iš /dev

Šiuos failus naudoja sisteminės programos, tačiau apie keletą jų verta žinoti plačiau.

Visi failai, kurie prasideda "tty", reprezentuoja terminalus. Pav., terminalą ttyO1 reprezentuoja spec. failas /dev/ttyO1. Jeigu jūs rašysite informaciją į šį failą, ta informacija bus perduota į atitinkamo terminalo ekraną.

Užduotis. Komanda who leidžia sužinoti, kokį terminalą naudoja jūsų kolega. Jusgalite naudoti cp (copy) komandą, kuri kopijuoja vieną failą į kitą. Jeigu failescary yra pranešimas kolegai, tai tą pranešimą galite perduoti į kolegos terminaloekraną komanda cp scary /dev/ttyO1. Be to jūs galite sužinoti jūsų terminaląreprezentuojančio spec. failo vardą komanda tty. Rezultatas: ?

Visuomet galima naudoti /dev/tty failą, kaip jūsų terminalo sinonimą. Jei programa rašo duomenis į failą, tai vietoje failo vardo nurodžius /dev/tty, duomenys bus išvedami į jūsų ekraną.

Naudingiausias /dev katalogo failas yra nuli. Sis failas reprezentuoja neegzistuojantį įtaisą. Skaitant iš /dev/null įtaiso, 'nieko' nenuskaitoma; rašant į /dev/null, rašymas 'atšaukiamas'. Pav., jeigu programa updatef iles atnaujina duomenis tam tikruose failuose ir darbo protokolą išduoda į ekraną, tai pagal komandą updatef iles > /dev/null ši programa atliks savo 'darbą', tačiau protokolo išdavimas bus ignoruotas.

4.7 Namų katalogai

Unix sistemos sprendimas - skirti kiekvienam vartotojui *Home Directory*. Tai su vartotojo *userid* asocijuotas ir vartotojo globojamas katalogas. Sistemos administratorius, registruodamas vartotoją, sukuria jam namų katalogą. Šio katalogo vardas saugomas sistemos slaptažodžių faile. Vartotojui prisijungus, jis automatiškai 'nuleidžiamas' į savo namų katalogą. Čia vartotojas turi teisę elgtis savo nuožiūra.

Tradiciškai namų katalogas kuriamos šakoje */usr*, tačiau tai nėra taisyklė. Paprastai šio katalogo vardas sutampa su '*userid*'. Dažnai administratoriai vartotojus skirsto į grupes, pav., *undergrad*, *grad*, *guests*, *staff* ir pan. ir *home* katalogus kuria atitinkamose šakose. Vartotojui prisijungus, jo aplinkos kintamasis *\$HOME* įgyja namų katalogo vardą. Be to, C-shell'as leidžia naudoti namų katalogo trumpinį *~* (tildė).

Užduotis. Koks jūsų namų katalogo vardas vardas? _

Pažiūrėkite *\$HOME* kintamojo reikšmę (komanda `echo $HOME`), arba pažiūrėkite trumpinio *~* reikšmę (komanda `echo ~`).

SKYRIUS 5 Failų prieinamumas

5.1 - Failų charakteristikos. Komanda l s	5-2
5.2 - Realūs bei efektyvūs vartotojų bei grupių identifikatoriai	5-6
5.3 - Failo prieinamumo nustatymas	5-8
5.4 - chmod komanda	5-9
5.5 - umask komanda	5-11
5.6 - chgrp, chown komanda	5-12

5.1 Failų charakteristikos. Komanda ls

Šiame skyriuje jūs susipažinsite su UNIX komandomis, skirtomis failų bei katalogų charakteristikų peržiūrai, išmoksite keisti bei nustatyti įvairias failų prieinamumo galimybes. Su kiekvienu failu yra surišta eilė charakteristikų, kurias galime sužinoti naudodami `ls` komandą. Jos išvedamų į ekraną charakteristikų kiekį nusako komandos opcijos. Pavyzdžiui, įvykdžius komandą:

Iš `-lgsF failo-vardas` matysime tokias charakteristikas:

```
-rwxrw-rw-      petringa      cust      512      Jan 30 00:14
```

1. 2. 3. 4. 5. 6.

1. Failo užimamų blokų skaičius
2. Failo prieinamumas – kas ir ką gali daryti su failu/katalogu
3. Failo ryšių kiekis
4. Failo savininko vardas
5. Failo savininko grupės vardas
6. Failo dydis baitais
7. Failo paskutinio modifikavimo data
8. Failo vardas

Antras laukas nusako failo tipą bei prieinamumą. Trečiame lauke yra nurodytas ryšių kiekis. Jis rodo, kiek vardų katalogų medyje yra surišta su tuo pačiu fiziniu failu. UNIX yra galimybė kreiptis į tą patį failą iš skirtingų vietų skirtingais vardais. Šie ryšiai yra sukuriami naudojant `In` komandą. Ketvirtas laukas rodo failo savininką. Kiekvienas UNIX failas turi savininką, juo yra vartotojas, sukūręs šį failą. Penktas laukas rodo grupę, kuriai priklauso failo savininkas. Septintame lauke yra nurodomas failo.5-2Failų prieinamumas

paskutinės modifikacijos laikas. Šį laiką naudoja koordinatorius `make`, `f ind` komanda ir kt. Failo vardas, kuris nurodomas aštuntame lauke, gali turėti iki 14 simbolių (iki 255 BSD UNIX). UNIX nereikalauja, kad failo vardas būtinai turėtų išplėtimą, tačiau kai kurios programos, pavyzdžiui C kalbos kompiliatorius reikalauja, kad failų vardai turėtų plėtinį ". c". UNIX draudžia suteikti failui vardą "." arba "..", nes šie vardai atitinka einamąjį arba tėvo katalogą.

Failo tipą nusako antro lauko pirmas simbolis, tai yra:

Simbolis	Prasmė
-	paprastas(regular) failas
d	katalogas
b	blok-oriėntuotas specialus failas
c	simbolinio tipo specialus failas
l	simbolinis ryšys
p	Vamzdis (?) (pipe)
s	šakutė (socket)

Užduotis:

a)Įvykdykite komandą

Iš `-al ..`

Kokius matote katalogus, kas jų savininkai, kokiai vartotojų grupei jie priklauso?

b) Įvykdysite komandą

`ls -al /dev`

Kokio tipo failai yra šiame kataloge?

c) Įvykdysite komandą

`ls -al /bin`

Kokio tipo ir dydžio failus matote?

Sekantys devyni simboliai charakterizuoja failo prieinamumą. Sistemos administratorius turi visas prieinamumo galimybes peržiūrėti vartotojų failus bei katalogus, nežiūrint į tai, kokios jos yra užduotos.

Lentelėje pateiktame pavyzdyje nurodytos tokios failo prieinamumo reikšmės:

`rw-rw-rw-`

Šie prieinamumo simboliai yra skirstomi į tris grupes po tris simbolius ir nusako priėjimo prie failo galimybes: pirmi trys simboliai galimybes vartotojui - failo savininkui, antri trys - vartotojams, kurie priklauso tai pačiai grupei kaip ir failo savininkas, paskutiniai trys - likusiems vartotojams. Čia `r` raidė žymi galimybę failą skaityti, `w` - failo rašymo galimybę, `x` - vykdymo galimybę. Brūkšnelis rodo, kad atitinkamos galimybės nėra.

Pavyzdžiui:

`rw-rw-r--` prieinamumo reikšmės žymi, kad failo savininkas turi teisę rašyti skaityti ir vykdyti šį failą, grupės vartotojai gali skaityti ir rašyti, o visi kiti - tik skaityti.

Visos šios galimybės yra kiek skirtingai traktuojamos priklausomai nuo failo tipo.

•Paprastam failui skaitymo galimybė reiškia, kad procesas gali failą skaityti, rašymo galimybė, - kad procesas gali keisti failo turinį, vykdymo galimybė, - kad procesas gali šį failą vykdyti.

•Katalogui skaitymo galimybė reiškia, kad procesas gali skaityti katalogą(tai yra perskaityti sąrašą failų, esančių tame kataloge). Rašymo galimybė reiškia, kad procesas

gali išmesti failus iš katalogo arba juos įkelti į katalogą. Vykdyimo galimybė reiškia, kad procesas gali prieiti prie failų esančių kataloge arba jo pakatologiuose.

•Specialiam failui skaitymo galimybė reiškia, kad procesas gali skaityti iš failo, naudodamas kreipinį `read()`. Analogiškai rašymo galimybė reiškia, kad procesas gali rašyti į failą, naudodamas kreipinį `write()`. Vykdyimo galimybė neturi prasmės.

Užduotis:

ls komanda išvedė tokius rezultatus:

```
drwxrw-r—  2  tomas  user1  512  Jan 30 05:14  kat1
```

```
drwxr—r—  2  petras  user2  512  Jan 30 08:14  kat2
```

Vartotojas Domas priklauso grupei `user1`, o vartotojas Elijus grupei `user3`. Kokius veiksmus gali atlikti šie vartotojai kataloguose `kati` bei `kat2`?

Užduotis:

`file` komanda atlieka failų identifikavimą. Duokite komandą

`file *`

Paskaitykite komandos `file` *Manual* puslapį.

5.2 Realus bei efektyvus identifikatoriai

Su kiekvienu vykdomu procesu yra surišami keturi identifikatoriai:

realus vartotojo identifikatorius **efektyvus vartotojo** identifikatorius **realus grupės** identifikatorius **efektyvus grupės** identifikatorius

Su šiais identifikatoriais irgi yra surištos tam tikros prieinamumo galimybės. Vartotojui prisijungiant prie sistemos startuoja `shelPo` procesas, kuris turi nustatytus vartotojo realų ir efektyvų identifikatorius, sutampančius su vartotojo identifikatoriumi, realų ir efektyvų grupės identifikatorius, sutampančius su vartotojo grupės identifikatoriumi. Kadangi failo prieinamumo reikšmės gali būti skirtingos vartotojui, grupei bei kitiems vartotojams, tai juos taikant yra atliekamas tikrinimas:

- jei proceso efektyvus vartotojo identifikatorius yra toks pats kaip failo savininko identifikatorius, tai yra pritaikomos vartotojui nustatytos prieinamumo galimybės,

- jei proceso efektyvus vartotojo identifikatorius skiriasi nuo failo savininko identifikatoriaus, bet efektyvus grupės identifikatorius sutampa su failo grupės identifikatoriumi, tai pritaikomos grupės prieinamumo galimybės,

- likusiais atvejais taikomos kitiems vartotojams skirtos failo prieinamumo galimybės.

Visos šios prieinamumo galimybės leidžia apsaugoti failus nuo nesankcionuoto priėjimo bei naudotis bendrais failais, jei vartotojai priklauso tai pačiai vartotojų grupei.

Vykdomiesiems failams UNIX sistemoje gali būti nustatytas specialus bitas, **s (the sët user/group id bit)**, liepiantis nustatyti vartotojo identifikatorių arba grupės identifikatorių. Jis atžymimas `s` raide vartotojo arba grupės laukuose vietoje simbolio `x`. Jei failui yra nustatytas šis bitas, tai failo vykdymo metu efektyvus vartotojo identifikatorius arba efektyvus grupės identifikatorius yra keičiamas į vykdomo failo savininko identifikatorių arba jo grupės identifikatorių, o tai suteikia vartotojui šio failo vykdymo metu tokias pačias teises kaip ir failo savininkui. Tai įgalina vartotoją keisti įrašą slaptažodžių faile, nors šio failo savininkas yra `root`.

Šie bitai kaip ir failo prieinamumo galimybės failams gali būti nustatomi naudojant `chmod` komandą.

Dar sykį prisiminkite failų prieinamumo nurodymus ir jų prasmę:

Permission	File	Directory
r read	read a file	list files in ...
w write	write a file	create file in ...
		rename file in ...
		delete file ...
x execute	execute a shell script	read a file in ...
		write to a file in ...
		execute a file in ... execute a shell script in ...

5.3 Failo prieinamumo nustatymas

Naujai kuriamiems failams prieinamumo galimybės nustatomos pagal nutylėjimą, tačiau vartotojas gali jas užduoti ir pats. Failams prieinamumo galimybės gali būti nustatomos ir keičiamos šiomis komandomis:

Komanda	Paskirtis
chmod	Keičia failo prieinamumo galimybes
umask	Nustato prieinamumo galimybes naujai kuriamam failui
chgrp	Keičia grupę, kuriai priklauso failas
chown	Keičia failo savininką

5.4 Komanda chmod

Failo prieinamumo galimybes galima keisti komanda

```
chmod -Rpakeitimas {failo-vardas}
```

R raktas rodo, kad nustatymas galios rekursyviai, t.y. visuose katalogo pakatalogiuose. Lauke *{failo vardas}* gali būti nurodomi failai bei katalogai, o "pakeitimu" yra nurodoma kokiems vartotojams ir kas yra nustatoma. Vartotojas žymimas vienu simboliu:

u	savininkas
g	grupe
o	kiti
a	visi

Šis simbolis skelbia, kokiems vartotojams yra nustatomas failo ar katalogo prieinamumas. Pats prieinamumas nusakomas raide:

r	skaityti
w	rašyti
x	vykdyti
s	nustatyti vartotoj o ar grupės identifikatorių

Prieinamumas gali būti nurodomi tiek aukščiau išvardytomis raidėmis, tiek ir skaičiais: jei atitinkama galimybė leidžiama yra 1, o jei ne - 0. Vientuku-nuliukų seka vėliau gali būti užrašoma kaip aštuntainis skaičius. Pavyzdžiui:

Simbolių keitimas skaičiais

rwX	r--	r-x	simbolinis
111	100	101	
7	4	5	
	745		skaitmeninis

5.6 Komandos `chown` bei `chgrp`

Naujų vartotojų grupių sukūrimą UNIX sistemoje vykdo sistemos administratorius. Sukūrus naują grupę, vartotojai gali būti paskelbiami šios grupės nariais. Vartotojas gali priklausyti kelioms grupėms.

Norint sužinoti, kokių grupių nariu yra vartotojas, jis gali panaudoti komandą

`groups [ vartotojo-identifikatorius ]`

Nenurodžius vartotojo-identifikatoriaus yra išvedamas grupių sąrašas vartotojui, įvykdžiusiam šią komandą.

Vartotojas gali pakeisti grupę, kuriai priklauso failas, įvykdydamas komandą

`chgrp -R grupės-identifikatorius {failo-vardas}`

vartotojui leidžiama pakeisti grupę tik tiems failams, kurių savininku yra jis pats, o sistemos administratoriui leidžiama keisti bet kurio failo grupę. Komandoje gali būti nurodytas failų sąrašas. `-R` raktas liepia keisti grupę rekursyviai visiems katalogo failams. Failo savininką gali keisti sistemos administratorius, naudodamas komandą

`chown -R naujas-vartotojas {failo-vardas}.`

Vietoje failo gali būti nurodomas ir katalogas. Kurdamas namų katalogą naujam vartotojui, sistemos administratorius visuomet privalo keisti šio katalogo savininką bei grupę, nes, administratoriui sukūrus naują katalogą, jis pradžioje ir priklausys administratoriui bei turės administratoriaus grupės identifikatorių. Tas pats bus ir su failais, kuriuos administratorius turi įkelti į vartotojo katalogą: `. cshrc . login .logout`. Taigi, administratorius privalo vykdyti `chown` bei `chgrp` komandas su

opcijomis `-R`, kad vartotojas būtų paskelbtas katalogo bei failų savininku, o grupė nustatyta tokia, kokia yra vartotojo grupė.

Jei vartotojas priklauso kelioms grupėms, tai jis gali pakeisti grupę, vykdydamas komandą

newgrp grupės-identifikatorius

Šiuo atveju bus sukuriamas naujas shell'o procesas, kuris turės naują grupės identifikatorių, o pradinis shell'o procesas lauks šio naujo proceso pabaigos.

SKYRIUS 6 Failų sistemos komandos

6.1 -	UNIX katalogų ir kelių klasifikacija	6-2
6.2 -	Absoliutus kelias iki einamojo katalogo	6-3
6.3 -	Naujų katalogų sukūrimas	6-3
6.4 -	Katalogo turinio peržiūrėjimas naudojant įvairias opcijas	6-4
6.5 -	Perėjimas į kitą katalogą	6-4
6.6 -	Failų perkėlimas į kitą katalogą	6-5
6.7 -	cat komandos teikiamos galimybės	6-6
6.8 •	Informacinių srautų perskirstymas	6-6
6.9 -	Failo turinio peržiūrėjimo komandos	6-8
6.10 -	Failų kopijavimas	6-9
6.11 -	Failų ir katalogų naikinimas	6-10
6.12 -	Failo spausdinimas	6-10
-	wc komanda	6-11

6.1 UNIX katalogų ir kelių klasifikacija

Katalogai (Directories): UNIX katalogai apjungti į medžio tipo struktūrą. Kataloguose saugomi žemesnio lygio katalogai ir failai.

Šakninis(Root) katalogas: yra medžio viršuje ir žymimas / simboliu.

Namų(Home) katalogas: jis dar vadinamas **login** katalogu. Sukuriamas kiekvienam sistemos vartotojui ir į jį patenkama automatiškai, prisijungus prie sistemos.

Einamasis (Current): katalogas, kuriame jus esate einamuoju momentu. Jis dar vadinamas darbiniu katalogu.

Keliai (pathnames): **absoliutūs (absolute)**, jei jie nurodyti šakninio katalogo atžvilgiu ir prasideda / simboliu. Pavyzdžiui, **/home/saule/staff**. Skirkite dvi / simbolio reikšmes - jis žymi šaknį ir kartu yra skyriklis.

santykiniai (relative), jie nurodomi einamojo katalogo atžvilgiu. Pavyzdžiui., jeigu einamasis katalogas yra **saule**, tai jo atžvilgiu gali būti nurodytas kelias **stud/os/fuksas**.

6.2 Absoliutus kelias iki einamojo katalogo

pwd (print working directory)

pwd komanda parodo pilną kelią nuo šaknies iki einamojo katalogo. Pasitikrinkite koks jūsų einamasis katalogas. Tai viena iš nedaugelio komandų, neturinti raktų (opcijų). Be to ši komanda (ji yra vidinė shell'o komanda) galioja visuose plačiai naudojamuose shelluose csh, tcsh, sh, ksh ir pan.

Kokiu atveju ši komanda neveiks?

6.3 Naujų katalogų sukūrimas

Komanda:

mkdir *katalogo_yardas*

Ši komanda einamajame kataloge sukuria katalogą.

Užduotys:

Namų kataloge sukurkite katalogus: Iab2, Iab3, Iab4, Iab5.

Koks sukurtų katalogų prieinamumas, kas jų savininkas, kokia grupė? p.s. pagal nutylėjimą (sitemos nustatytas) prieinamumas katalogams yra 777, o failams 666.

Iš šių reikšmių yra minusuojama umask reikšmė ir tokiu būdu gaunama naujai sukurtų objektų prieinamumo reikšmė.

Uždrausite sukurtų katalogų Iab2, Iab3, Iab4, Iab5 prieinamumą kitiems vartotojams, išskyrus jūsų grupę.

6.4 Katalogo turinio peržiūrėjimas naudojant įvairias opcijas

ls katalogas

Ši komanda leidžia, naudojant įvairius raktus (opcijas) gauti platesnę arba siauresnę informaciją apie failus bei katalogus, esančius nurodytame kataloge.

Užduotis:

Naudodami man susipažinkite su *ls* komanda ir jos pateikiamais rezultatais. *ls* komanda peržiūrėkite einamojo katalogo turinį. Panaudokite įvairias opcijas. Susipažinkite su komanda *dir*.

Žiūrėti failų tipus ir i-node numerius [*ls -Fi*]

Žiūrėti kiekvieną pakatalogį ir jų turinius rekursiškai [*ls -R /kernai*]

6.5 Perėjimas į kitą katalogą

cd katalogas

cd komanda naudojama perėjimui į kitą katalogą. Pavyzdžiui įvykdžius

<i>cd labi</i>	<i>labi</i> taps einamuoju katalogu,
<i>cd ..</i>	perkels medžiu aukštyn,
<i>cd ../..</i>	perkels medžiu aukštyn per du lygius,
<i>cd</i>	grąžins jus į <i>home</i> katalogą,
<i>cd /</i>	perkels į šakninį katalogą.

Šioje komandoje esantys du taškai žymi perėjimą katalogų medžiu vienu lygiu aukštyn, vienas taškas naudojamas einamojo katalogo nurodymui, *~* ženklas ir po jo sekantis

varotojo *login vardas* leidžia pereiti į nurodyto varotojo namų katalogą.

Užduotis:

Pereikite į katalogą lab2

Įvykdykite komandas

`pwd`

`ls -a`

`dirs`

Kokia tai komanda, vidinė ar išorinė?

`dirs -l`

6.6 Failų perkėlimas į kitą katalogą

`mv` komanda naudojama perkelti failą arba pakeičiant katalogo(failo) vardą.

Perkelti failą iš einamojo katalogo į pakatalogį vykdoma komanda:

`mv failas pakatalogis/`

Šiuo atveju failas bus perkeltas tuo pačiu vardu. Perkelti galima keisti failo vardą:

`mv failas1 pakatalogis/ failas2`

Katalogas, į kurį vykdomas failo perkėlimas gali būti nurodomas nurodant absoliutų kelio vardą.

Užduotis:

Perkelkite failus iš savo namų katalogo į katalogą lab2

6.7 cat komandos teikiamos galimybės

Sistemoje UNIX naują failą galima sukurti įvairias būdais, vienas paprastesnių yra jo įvedimas iš klaviatūros prieš tai surinkus cat komandą:

`cat > naujo-failo-vardas` Po to yra įvedami failo įrašai.

Įvedimo pabaiga nurodoma ^D klavišais.

Užduotis:

Pereikite į katalogą lab3.

Komanda cat sukurkite naują failą vardu pavyzdi, kuriame talpinsime 3 eilutes žemiau pateikiamo teksto:

```
cat > pavyzdi
tai pirma eilute
tai antra eilute
ir trečia eilute
^D
```

cat komanda dažnai naudojama tekstinių failų peržiurai. Tai atliekama surinkus komandą

`cat failo-vardas`

Be to, naudojant cat komandą galima apjungti failus ir jų turinys parodomas ekrane. Failų apjungimas yra tikroji šios komandos paskirtis.

Užduotis:

Sukurkite failą pavyzd2, į kurį įrašykite sekantį tekstą:

O tai ketvirtoji eilutė.

^D

Atlikite failų apjungimą, naudodami komandą:

```
cat pavyzd1 pavyzd2
```

6.8 Informacinių srautų perskirstymas

UNIX sistemoje komandos pagal nutylėjimą duomenis ima iš klaviatūros (standartinis įvedimo failas), o komandų įvykdymo rezultatai yra nukreipiami į ekraną (standartinis išvedimo failas). Komandų vykdymo rezultatai gali būti nukreipiami į failą, panaudojant simbolį `>`. Tokią konstrukciją jau panaudojome kurdami failus pavyzdi pavyzd2. Dabar nukreipkime failų pavyzdi ir pavyzd2 apjungimo rezultatą į failą catout:

```
cat pavyzdi pavyzd2 > catout
```

Pastaruoju atveju komandos įvykdymo rezultatų ekrane nematysim, bus sukurtas naujas failas catout. Jį galima peržiūrėti komanda:

```
cat catout
```

Pastaba: Jei nukreipiate standartinį išvedimą į failą

komanda > failo-vardas

tai:

- komandos įvykdymo rezultatų ekrane nebus, jie bus failo *failo-vardas*;
- jei tokio failo *failo-vardas* dar nebuvo, jis bus sukurtas;
- jei failas *failo-vardas* jau buvo, jo turinys bus išgadintas.

Simbolis `>` nukreipia standartinį išvedimą (terminalo ekranas) į nurodytą failą. Atitinkamai norint įvesti duomenis ne iš klaviatūros, o imti iš failo, yra naudojamas simbolis `<`. Todėl komanda

komanda < failas1 > failas2 žymės tai, kad duomenys bus imami iš failo

failas1, o rezultatai talpinami failo *failas2*.

6.9 Failo turinio peržiūrėjimo komandos

Tai: more, head ir tail komandos.

more komanda, taip pat kaip ir cat komanda, parodo failo turinį ekrane. Ji naudotina ilgų failų peržiūrai, kadangi more komanda išveda informaciją į ekraną puslapiais:

```
more failo-vardas
```

Pereiti prie kito puslapio galima spaudžiant tarpo klavišą. Šią komandą galima naudoti kitų komandų išvedamų rezultatų peržiūrai. Pavyzdžiui:

```
ls -al | more
```

Panaudojus komandą head galima pamatyti pirmas 10 failo eilučių:

```
head failo-vardas
```

Panaudojus komandą tail galima pamatyti paskutines 10 failo eilučių:

```
tail failo-vardas
```

Užduotis:

Peržiūrėkite turinį su komanda cat ir more.

Panaudokite komandas head ir tail katalogo /bin turinio peržiūrėjimui.

6.10 Failų kopijavimas

Failo kopijavimui naudojama komanda `cp`. Įvykdžius komandą:

```
cp failas1 failas2
```

bus gauta failo `failas1` kopija. Aplamai reikėtų prisiminti bendrą katalogų kopijavimo komandos formą

```
cp -r katalogas1 katalogas2
```

Komanda kopijuoja *katalogas1* ir visa kas jame yra į katalogą *katalogas2*.

Pastarasis katalogas sukuriamas, jeigu jo nėra. Jeigu *katalogas2* yra, tai jame bus sukurtas pakatalogis *katalogas1*.

Vietoje kopijavimo komandų dažnai naudojama failų ir katalogų surišimo komanda `ln`.

6.11 Failų ir katalogų naikinimas

`rmdir` komanda pašalina tuščius katalogus iš einamojo katalogo,

```
rm -opcijos failas
```

ir jos opcijos leidžia naikinti failus. Reikia atsiminti, kad atstatyti išmestus katalogus arba failus galima tik tuo atveju, jei sistemos administratorius yra atlikęs atminties išsaugojimą.

Ši komanda naikina failą ir jo turinį. Įvykdysite komandą:

6.12 Failo spausdinimas

Norint failą spausdinti popieriuje, naudojama komanda `lpr` (`lp`) :

`lpr failo-vardas`

Spausdinimas galimas tik esant pajungtam spausdintuvui.

`lpq` komanda naudojama printerio eilės peržiūrėjimui: `lpq`
Komanda `lprm` naudojama norint išmesti tam tikrą užduotį iš spausdinimo eilės.
Užduoties numerį galima sužinoti komanda `lpq`, o po to naudoti komandą
`lprm uzduoties-nr`

6.13 `wc` komanda

Šios komandos pagalba galima suskaičiuoti eilučių, žodžių ir simbolių kiekį viename arba keliuose failuose:

`wc failas1 failas2 ... <CR>`

Kontroliniai klausimai.

- Kaip galima sužinoti, kokie katalogai yra einamajame kataloge?
 - Kaip sužinoti, ar failas yra vykdomasis?
 - Kaip atžymimi specialūs failai, ryšiai (link)?
 - Kaip peržiūrėti kito katalogo turinį, nekeičiant einamojo katalogo?
 - Kaip perkelti visus failus į kitą katalogą?
 - Kaip perskirstyti įvedimo/išvedimo duomenis?
 - Kaip įrašyti paskutines failo eilutes į naują failą?
- Kaip išnaikinti katalogą su visu jo turiniu?

SKYRIUS 7 vi redaktorius

7.1 - Trumpai apie vi	7-2
7.2 • Redaktoriaus darbo režimai	7-3
7.3 - vi komandos:	
kursoriaus valdymas (komandinis režimas)	7-6
7.4 - Papildomos kursoriaus valdymo komandos:	7-7
7.5 - vi komandos: teksto išmetimas	7-8
7.6 - vi komandos: koregavimas	7-9
7.7 - vi komandos: veiksmo atstatymas	7-10
7.8 - vi komandos: eilučių sujungimas	7-11
7.9 - Redagavimo pabaiga	7-12
7.10 - Eilučių numeriai	7-14
7.11 - vi komandos: markeriai	7-15
7.12 - vi komandos: kiti įvedimo režimai	7-16
7.13 - vi komandos:	
išpiovimas, kopijavimas, perkėlimas	7-17
7.14 - vi komandos: kontekstinė paieška	7-19

7.1 Trumpai apie vi

vi (Visual Interactive) - tai vienas iš populiariausių UNIX tekstų redaktorių. Visų pirma jį rasime visose UNIX sistemose. Juo galima pasinaudoti tais momentais kai kiti tekstų redaktoriai yra neprieinami (pavyzdžiui perinstaliuojant sistemą). Antra puiki šio redaktoriaus savybė yra ta, kad jis naudoja mažai atminties, o tai yra svarbu esant perkrautam tinklui. Trečia - vi komandos yra nusakomos raidėmis-skaičiais, todėl jį galima naudoti bet kokiame terminale ar darbo stotyje. Istoriškai vi redaktorius yra UNIX redaktoriaus ex modifikacija. Vi redaktoriaus komandos yra jautrios registrui (reikia atidžiai naudoti mažąsias ir didžiąsias raides).

Redaktorius iškviečiamas taip: vi

failo-vardas<CR>

Čia *failo-vardas* - tai vardas failo, kurį norime redaguoti. Ekrane pasirodys nurodyto failo eilutės, arba ekranas liks tuščias, jei kuriamas naujas failas. Failo vardo nenurodžius, vi redaktorius užklaus jo jums baigus redagavimą.

Užduotis:

Pradedame kurti failą vardu sample. Įveskite komandą:

vi sample <CR>

Ekranas išvalomas, nes kuriamas naujas failas. Vi redaktorius naudoja ~ simbolį tuščių eilučių pažymėjimui bei kursorium atžymi tą teksto vietą, kuri bus veikiamą tolimesnių komandų ar įvedamo teksto. Lango apačioje yra pranešimų eilutė, kurioje nurodomas failo vardas bei būvis.

7.2 Redaktoriaus darbo režimai:

Redaktorius turi du darbo režimus: komandinį bei teksto įvedimo.

- **komandinis.** Naujai iškvietus redaktorių, jis visuomet būna komandiniame režime, o iš teksto įvedimo režimo į komandinį režimą yra pereinama nuspaudus Esc klavišą. Komandiniame režime redaktorius priima, interpretuoja ir vykdo iš klaviatūros pateikiamas komandas, keičia kursoriaus padėtį, išpiauna arba kopijuoja tekstą, vykdo teksto paiešką.
- **teksto įvedimo.** Šiame režime įvedamas tekstas. Iš komandinio režimo į teksto įvedimo režimą pereinama nuspaudus vieną iš lentelėje nurodytų klavišų:

klavišas	Veiksmas
a	įterpti tekstą į dešinę nuo kursoriaus,
A	įterpti tekstą eilutės gale,
i	įterpti tekstą į kairę nuo kursoriaus,
I	įterpti tekstą eilutės pradžioje,
o	įterpti eilutę žemiau einamosios eilutės,
O	įterpti eilutę prieš einamąją,
R	rašoma ant teksto.

Vi redaktorius niekaip neindikuoja, kokiame režime jis randasi. Paspaudus vieną iš lentelėje nurodytų klavišų jokio pokyčio ekrane nesimato, redaktorius persijungęs į teksto įvedimo režimą ir laukia jūsų įvedamo teksto. Teksto įvedimo metu galima naudoti **backspace** klavišą paskutinio įvesto simbolio taisymui, tačiau negalima judėti ekrane naudojant kursorių valdančius klavišus, nes juos spaudžiant juos atitinkantis kodas įterpiamas į tekstą.

Be **backspace** klavišo, įvesto teksto taisymui galima naudoti ir šių klavišų kombinacijas:

CTRL-W	Ištrina prieš tai einantį žodį.
CTRL-U	Trina einamąją eilutę.
GR	Pereina į naują eilutę.

Baigus teksto įvedimą į komandinį režimą grįžtama nuspaudus klavišą Esc. Jei jus nesate tikri, kad esate sugrįžę į komandinį režimą, galite pakartotinai nuspausti Esc klavišą ir jei jūs jau ir taip buvote komandiniame režime, tai vi redaktorius garsiniu signalu informuos jus apie tai.

Užduotis:

Klaviatūroje surinkite:

```
:set showmode <CR>
```

Ekrane jokio pokyčio nesimato, nes būnant komandiniame režime indikatorius nesimato, bet būvio indikatorius atsiras perėjus į įvedimo režimą. Kai komandiniame režime jūs spaudžiate ":" klavišą, tai leidžia jums susirišti su ex redaktoriaus komandomis ir jas efektyviai panaudoti vi redaktoriaus aplinkoje.

Užduotis:

Klaviatūroje surinkite: i

Apatiniam dešiniam kampe vi indikuos, kad jūs randatės įvedimo režime. Įveskite sekantį tekstą, nepamiršdami spausti <CR> kiekvienos eilutės gale.

Pradedame tekstu :

Vienas, vienas, vienas, o gal trys?

Du ir trys - tarsi penki...

1997 metai sausio pabaiga

Laikas bėga labai greitai ...

Surinkite šį tekstą ir paspauskite Esc klavišą - taip jūs vėl grįšite į komandinį režimą.

eilutės režimas (Line mode).

Komandiniame režime nuspaudus vieną kurį nors klavišą : arba / arba ?, kursorius nušoka į paskutinę ekrano eilutę, išveda į ekraną įvestą simbolį, ir po šio simbolio gali būti duodama komanda, po kurios įvykdymo redaktorius lieka komandiniame režime. Eilutės režimą galima naudoti vykdant paieškos, keitimo ar failo išsaugojimo komandas.

7.3 vi komandos:

kursoriaus valdymas (komandinis režimas)

Jums dažnai gali tekti keisti kursoriaus padėtį faile. Ji keičiama būnant komandiniame režime šiomis keturiomis komandomis:

Klavišas	Veiksmas
j arba ↓	Perkelti kursorių viena eilute žemyn.
k arba ↑	Perkelti kursorių viena eilute aukštyn.
h arba ←	Perkelti kursorių vienu simboliu į kairę.
l arba →	Perkelti kursorių vienu simboliu į dešinę.

Vi redaktorius naudoja savus sakinio, paragrafo apibrėžimus:

Sakiniu vadinama simbolių seka tarp dviejų sakinio skyrybos ženklų: taško(.), klaustuko(?) arba šauktuko (!). Tuščia eilutė taip pat baigia sakinį.

Eilute vadinama simbolių seka tarp dviejų <CR> paspaudimų.

Paragrafu laikoma seka eilučių tarp kurių nėra tuščių eilučių.

7.4 Papildomos kursoriaus valdymo komandos:

b	'kelti kursorių atgal per Meną žodį
w	Perkelti kursorių tolyn per vieną žodį,
e	Einamojo/sekančio žodžio pabaiga,
0 (nulis) arba ^	eilutės pradžia,
\$	eilutės pabaiga.
(	Einamojo arba prieš tai buvusio sakinio pradžia.
)	Sekančio šakinio pradžia.
{	Einamojo paragrafo pradžia.
}	Einamojo paragrafo pabaiga.
M	Ekrano vidurinė eilutė.
H	Perkelti kursorių į ekrano viršų.
L	Perkelti kursorių į paskutinę ekrano eilutę.

Kai kurios redagavimo komandos gali būti taikomos eilučių **seka**i. Galima naudoti vieną iš šių lentelėje nurodytų būdų.

seka	Išrenkama
1,\$	visos failo eilutės
1,.	nuo pirmos iki einamosios eilutės (ją įskaitant)
.,\$	nuo einamosios eilutės (ją įskaitant) iki failo galo
.-3	viena eilutė, kuri yra prieš tris eilutes nuo einamosios

7.5 vi komandos: teksto išmetimas

Kartais redagavimo metu jums gali tekti išmesti dalį teksto. Kursorius turi rasti ties pirmu jūsų išmetamos grupės simboliu ir tada galima naudoti vieną iš komandų:

išmetamas dydis	Komanda
simbolis	Kursorius turi būti ties simboliu ir spaudžiamas klavišas x
eilutė	Kursorius turi būti bet kur toje eilutėje ir naudojama komanda dd
žodis	Kursorius turi būti žodžio pradžioje ir naudojama komanda dw
iki eilutės galo	Kursorius turi būti bet kur toje eilutėje ir naudojama komanda D
eilučių seka	:<seka>d

Pastaba: šiose komandose galimi kartotinumai. Pvz., komanda 2dd pašalina dvi eilutes. Antroji komandos raidė nusako teksto dalį, kuriai taikoma komanda, pavyzdžiui d\$ veiks taip pat kaip komanda D - tai yra išmes iki eilutės galo.

Užduotis:

kursorių valdančiais klavišais ateikite ties "o" raide antroje eilutėje ir naudodami komandą x ištrinkite raidę o.

Atveskite kursorių ties žodžio "**tekstu**" pradžia. Komanda **dw** ištrinkite šį žodį. Ištrinkite visą pirmąją eilutę naudodami komandą dd.

7.6 vi komandos: koregavimas

Dažnai tenka keisti tekste raides, žodžius ar visas eilutes. Atliekant šio tipo pakeitimus kursorių reikia atvesti ties keičiamo teksto pradžia, surinkti vieną iš keitimo komandų, o po to tekstą, kuriuo keičiamas teksto fragmentas, ir keitimą baigti paspaudus Esc klavišą (išskyrus komandą r).

komanda	Veiksmas
cw	Keičiamas žodis, ženklu vi pažymi keičiamos porcijos pabaigą.
C	Eilutė perrašoma iki pabaigos.
r	Vienas simbolis keičiamas kitu. Esc spausti nereikia, nes pasilieka komandiniame režime.
R	Perrašomi simboliai pradedant nuo einamosios kursoriaus pozicijos.
s	Vienas simbolis keičiamas vienu arba keliais simboliais
S	Einamoji eilutė keičiama nauja eilute.

Šiose komandose irgi galima naudoti antrąją raidę, kuria galima nusakyti keičiamo teksto porciją. (c\$).

Užduotis:

- Atveskite kursorių ties žodžio "tarsi" raide V ir pasinaudodami komanda r pakeiskite raidę "r" į raidę "i". Redaktorius išliks komandiniame režime.
- Išmeskite raides "si", likusias šiame žodyje (komanda x).
- Pakeiskite žodį "sausio" į žodį "vasario" - Tam atveskite kursorių ties žodžio sausio pradžia, surinkite komandą cw, po to žodį "vasario" ir paspauskite Esc, tuogrįždami į komandinį režimą.
- Pakeiskite tekstą "labai greitai" tekstu "kaip vanduo" - tam atveskite kursorių ties raide "P, paspauskite C ir rinkite tekstą "kaip vanduo".
- Baigę - nuspauskite Esc klavišą. Vėl grįšite į komandinį režimą.

7.7 vi komandos: veiksmų atstatymas

Kartais jus atsitiktinai galite įvykdyti ne tą komandą arba išmesti tekstą, kurį nereikėjo išmesti. Vi redaktorius leidžia atstatyti paskutinį veiksmą su undo komanda, kuri atliekama, paspaudus u klavišą. Tačiau vi redaktorius gali atstatyti tik paskutinį veiksmą.

Užduotis:

Atveskite kursorių ties eilute 1997 metai... ir surinkite komandą :

dd.

Eilutė dings. Atstatykite ją :

u

Eilutė vėl pasirodys ekrane. Paspauskite u kelis kartus. Kiekvieną kartą eilutė dings arba pasirodys, nes kiekvienas u paspaudimas atstato prieš tai vykdytos komandos veiksmus. Baikite šį režimą palikdami ekrane eilutę 1997....

7.8 vi komandos: eilučių sujungimas

Kartais jums gali prireikti sujungti kelias eilutes į vieną. Naudojama komanda:

J

Ji .sujungia einamąją eilutę su sekančia eilute.

Užduotis:

Prieš eilutę "Du ir trys.." įterpsime dvi naujas eilutes. Atveskite kursorių ties eilute "Du ir trys..", ties raide D ir įvykdysite komandą :

O

Redaktorius pereis į įterpimo režimą ir lauks iš jūsų įvedamo teksto. Įveskite:

Tai dar viena eilutė

o taip pat

Primename, kad eilutės gale reikia spausti <CR>, o įvedimą reikia baigti nuspaudžiant Esc klavišą. Ateikime ties raide "T" ir įveskite komandą :

J

Įvestos eilutės bus sujungtos į vieną.

7.9 Redagavimo pabaiga

Baigti redagavimą galima įvykdant komandą `ZZ` (komandiniame režime). Tada redaguojamame faile bus išsaugoti visi pakeitimai. Galima baigti darbą ir kitaip, eilutės režime (:) surinkus vieną iš komandų:

klavišas	Veiksmas
<code>w</code>	Išsaugoti pakeitimus ir tęsti redagavimą
<code>q!</code>	išeiti, neišsaugant pakeitimų
<code>wq</code>	Įrašyti ; failą ir išeiti iš vi
<code>w <vardas></code>	Išsaugoti failą nauju vardu vardas

Išsaugant pakeitimus ir išeinant iš redaktoriaus, yra pranešama į kokį failą buvo įrašyta bei kiek eilučių/simbolių yra įrašyta.

Užduotis:

Išeikite iš redaktoriaus išsaugodami visą suvestą tekstą. Tai atlikite surinkdami: `:wq`

Pakartojimas

Užduotis:

Iškvieskite vi redaktorių tolimesniam failo **sample** tvarkymui:

vi sample

Vėl įjunkime režimo rodymą:

:set showmode

Kursorius rasis teksto pradžioje, o ekrane matysit surinktą tekstą:

Vienas, vienas, vienas, o gal trys?

Tai dar viena eilute o taip pat

Du ir trys - tai penki...

1997 metai vasario pabaiga •

Laikas bėga kaip vanduo...

Įveskite komandą žodžio naikinimui:

dw

Atstatykite išmestą tekstą:

u

Atveskite kursorių ties skaičiais 1997, už skaičiaus 7 ir surinkite:

5iBRAVO ! Esc

Prieš Esc palikite tarpą. Žodis BRAVO bus įdėtas 5 kartus.

7.10 Eilučių numeriai

Daugelis vi komandų naudoja eilučių numerius. Galima tekstą iškirpti, kopijuoti naudojant eilučių numerius. Galima taip pat nušokti į tam tikrą vietą tekste. Eilučių numeriai naudingi ir ieškant klaidų programos tekste, nes dažnai transliacijos eigoje yra pranešama, kad rasta klaida tam tikroje eilutėje.

Užduotis:

Ijungsim eilučių numerių rodymą:

:set number

Vi redaktorius kairiam krašte parodys eilučių numerius.

G komanda leidžia jums nušokti į norimą eilutę jūsų faile. Pradžioje yra rašomas eilutės numeris, o už jo raidė G. Jei eilutės numeris nenurodomas, tai vi redaktorius nušoks į failo pabaigą. Į G leis jums nušokti į pirmą failo eilutę.

Užduotis:

Būdami komandiniame režime pereikite nuo vienos eilutės prie kitos:

2G

1G

G

atsidurti pradžioje eilutės, turinčios tą markerį, vykdoma komanda **'raidė**.

Užduotis:

Atveskite kursorių ties žodžio "vanduo" pradžia. Čia įdėsime markerį v:

mv

Atveskite kursorių ties žodžiu "1997". Čia įdėsime markerį k:

mk

Peršokime tekstu, naudodami markerį:

'v

7.12 vi komandos: kiti įvedimo režimai

Pakartosime kitus teksto įvedimo režimus, vykdomus komandomis a A o O

a - liepia įterpti įvedamą tekstą po einamojo simbolio,

A - liepia įterpti įvedamą tekstą eilutės gale,

o -, liepia įterpti įvedamą tekstą naujoje eilutėje po einamosios,

O - liepia įterpti įvedamą tekstą naujoje eilutėje prieš einamąją.

Užduotis:

Atveskite kursorių po pirmo žodžio pirmoje eilutėje. Naudodami komanda a įterpsime simbolius +**** +

Grįžkite į komandinį režimą.

Naudodami komandą O prieš antrą eilutę įterpsime tokią eilutę:

Sutems tamsi ...

Grįžkime į komandinį režimą.

Ketvirtos eilutės gale, naudodami komandą A prirašykime simbolių

Grįžkime į komandinį režimą.

7.13 vi komandos: išplovimas, kopijavimas, perkėlimas

Redaguojant tekstus dažnai tenka išplauti, kopijuoti tekstą, perkelti jį į kitą vietą. Pradžioje tekstas yra įkeliamas į laikiną atmintį, o po to perkeliamas iš jos į naują vietą dokumente. Vi redaktorius naudoja devynis numeruotus buferius bei "undo" buferį, kuris saugo paskutinius išmetimus. Atliekant sekantį išmetimą "undo" buferio turinys keliaus į pirmą buferį, pirmo - į antrą, o devinto - bus prarastas. Vi redaktorius taip pat turi dvidešimt šešis raidėmis a-z įvardintus buferius. Buferiai yra naudingi saugant informacijos blokus, kai informacija yra iškerpama arba kopijuojama. Buferių turiniai nekinta, kol į juos nėra įrašoma nauja informacija. Bet, kaip ir markerių atveju, atsijungus nuo redaktoriaus, buferių turiniai bus prarasti.

Kopijavimo komandose galima panaudoti eilučių numerius. Komanda **m** perkelia tekstą jį iškirpdama, o komanda **t** - atlieka teksto kopijavimą:

:nr1mnum

Ši komanda perkels eilutės nrl turinį po eilute su numeriu num.

:nrl,nr2mnum

Ši komanda perkels eilučių nrl - nr2 turinį po eilute su numeriu num.

:nrltnum

Ši komanda kopijuos eilutės nrl turinį po eilute su numeriu num.

:nrl,nr2tnum

Ši komanda kopijuos eilučių nrl - nr2 turinį po eilute su numeriu num.

Užduotis:

Nukopijuokite pirmą eilutę po paskutine teksto eilute.

Perkelkite antros eilutės turinį po trečia eilute.

Vi redaktorius vykdydamas bet kurią iš išmetimo komandų automatiškai keičia buferių turinius. Jei yra vykdoma komanda "yank"(įkėlimas į buferius) ir buferis nėra nurodomas, tai redaktorius naudos automatinius buferius. Tokiu atveju kursorius yra nuvedamas į norimo teksto pabaigą, nurodomas buferis (raide, skaičiumi arba nutylint),

nurodomas kartotinumą pagal reikalą, pavyzdžiui 5 žodžiai ar eilutės, raidė d (iškirpimui) arba raidė y (įkėlimui) ir viena iš raidžių, naudojamų nusakant kursoriaus judesį (b e w ...). Paskutinės raidės nusakys teksto bloką, o jei jų nėra, tai bus imama eilutė.

Norint iš buferio tekstą įkelti į dokumentą, reikia pradžioje atvesti kursorių į tą poziciją, į kurią bus keliamas tekstas. Imant tekstą iš įvardinto buferio, jis yra nurodomas taip: "raidė. Neįvardinus bus imamas automatinio buferio turinys. Pats įkėlimo veiksmas atliekamas naudojant komandas p arba P. Komandos p atveju tekstas iš buferio bus įkeliamas už einamojo simbolio, o komandos P atveju - prieš einamąjį simbolį. Pavyzdžiui:

"ed'x Čia eilutėje iki markerio x esantis tekstas bus išplautas ir įdėtas į buferį e.

"ep

Bus į dokumentą įkeliamas buferio e turinys.

Užduotis:

Pasinaudodami buferiais atlikite išpiovimo bei perkėlimo į kitą eilutę veiksmus su įvairiais teksto fragmentais.

Išbandykite y bei p arba P komandas su įvairiais teksto fragmentais.

7.14 vi komandos: kontekstinė paieška

Komandiniame režime nuspausti / arba ?, o po jų nurodyti ieškomą kontekstą:

/kontekstas - ieškos nurodytos simbolių sekos faile (pirmyn), **^kontekstas** - ieškos nurodytos simbolių sekos faile (atgal),

n - surasti ankstesnę simbolių seką ankstesne kryptimi,

N - surasti ankstesnę simbolių seką, pakeitus paieškos kryptį.

Panaudojus paiešką galima pakliūti į tą teksto eilutę, kuri sutaps su ieškomu šablonu.

Vi redaktorius taip pat leidžia atlikti ne tik paiešką, bet suradus sutapimą ir pakeisti vieną simbolių seką kita seka. Keitimą žymi komanda s:

:num1,num2s/sena_seka/nauja_seka

Čia num1, num2 - tai eilučių numeriai. Jie užduoda teksto dalį paieškos/keitimo komandai. Jei numeriai nenurodyti paieška/keitimas lies tik einamąją eilutę. Keitimas lies tik pirmą sutiktą sutapimą. Norint pakartoti tai einamojoje eilutėje reikia naudoti komandą &, o kitoje eilutėje:

:numeris&

Kitoje eilučių grupėje:

:num1,num2&

Užduotis:

Išbandykite paieškos, paieškos/keitimo komandas savo faile.

Išeikite iš redaktoriaus vi išsaugodami sukurtą failą.