

# ATK tähtitieteessä

765332A-01 – osa Tähtitieteen tutkimusprojekti 1 -kurssia!

Kurssista päävastuussa prof. Heikki Salo (heikki.salo@oulu.fi),  
toimisto MA305.

Aiheet:

- Linux-työasemat.
- IDL-ohjelmointi.
- LaTeX-dokumenttien luominen.

Arvostelu: hyväksytty/hylätty.

Luentokalvot, esimerkkiohjelmat yms. materiaali ovat saatavilla  
yliopiston wikissä:

ATK-osuuden jälkeen alkaa varsinainen tutkimusprojekti. Lisätietoa  
myöhemmin.

# Aikataulu

Harjoitukset/luennot: YL 124:

19.04. ke 08.15-11.00

20.04. to 12.15-15.00

25.04. ti 12.15-15.00

26.04. ke 08.15-11.00

03.05. ke 08.15-11.00

08.05. ma 09.15-12.00

## Varoitus!

Tämä ei ole varsinainen ohjelmointikurssi!

Tämän kurssin tarkoituksena on esitellä työkaluja, jotka ovat  
spesifisiä tähtitieteessä, ja joita ei välttämättä käsitellä muilla  
ATK-kursseilla.

Nämä ovat niitä työkaluja, joita käytetään tähtitieteen kurssien  
harjoitustöissä tai demonstraatioissa.

## Käyttöjärjestelmistä

Wikipedia:

Käyttöjärjestelmä on keskeinen tietokoneen ohjelmisto. Se mahdollistaa muiden ohjelmien toiminnan. Se hallinnoi tietokoneen resursseja, luo laitteiston yksityiskohdista riippumattoman operointialustan ja [järjestelmäkutsut](#). Järjestelmäkutsuihin kuuluu korkeamman tason käsitteitä, kuten [tiedostojärjestelmä](#), [virtuaalimuisti](#), [muistinhallinta](#) ja [verkkoprotokollat](#) sekä niihin liittyviä operaatioita ja erinäisiä käytäntöjä.

Kone käynnistetään → käyttöjärjestelmä tarkistaa laitteiston, käynnistää joitakin palveluja ja jää odottamaan komentoja.

Käyttöjärjestelmää voi käskyttää antamalla komennot komentoriviltä tai käyttämällä graafista käyttöliittymää.

Useimmat käyttöjärjestelmät ovat moniajavia: käyttäjän näkökulmasta useampi ohjelma on samaan aikaan käynnissä.

Useimmat käyttöjärjestelmät ovat monen käyttäjän käyttöjärjestelmiä.

Samassa koneessa voi olla useita käyttöjärjestelmiä, jolloin ns. bootladeria käyttäen voi valita käynnistettävän käyttöjärjestelmän.

Yleisin kombinaatio Windows & Linux.

Käyttöjärjestelmää on mahdollista myös ajaa "toisen alla" ns. virtuaalikoneessa.

Yleisimpiä käyttöjärjestelmiä tällä hetkellä ovat Microsoft Windows, MacOS X ja Linux.

Linuxista on sekä ilmaisia että kaupallisia versioita.

## Tietoturva

Tietoturva tarkoittaa montaa asiaa:

- tiedon säilymisen varmistaminen
- luottamuksellisen tiedon varjelu ulkopuolisilta
- tietokoneressurssien turvaaminen

Tiedon säilymistä uhkaavat laiterikot, tallennusmedian vanheneminen ja ennen kaikkea yleinen tunarointi.

Tärkeistä tiedostoista kannattaa ottaa säännöllisesti varmuuskopioita.

Varmuuskopioinnissa voi käyttää kiintolevyn sisällön osittaista tai täydellistä kopiointia esim. muistitikulle, ulkoiselle kiintolevylle tai optiselle levyille.

Monilla palvelinkoneilla esim. käyttäjien kotihakemistoista otetaan ylläpidon toimesta säännöllisesti varmuuskopiot.

Varmuuskopioiden laatu on syytä tarkistaa säännöllisesti!

Tähtitieteilijät käsittelevät harvoin valtiosalaisuuksia ja “teollisuusvakoilukaan” ei ole arkipäivää.

Luottokorttitietojen tai erilaisten henkilökohtaisten tiedostojen päätyminen vääriin käsiin ei ole kuitenkaan kiva asia.

Tietomurto voi uhata sekä datan säilyvyyttä että yksityisyyttä.

Tietomurron haitat voivat koskettaa myös tietokoneresursseja itseään:

- murrettua konetta voidaan käyttää uusien tietomurtojen tekemiseen.
- murretusta koneesta voidaan tehdä laittoman materiaalin välityskanava.

Tietoturvariskejä voi torjua monin tavoin.

- Salasanat ovat todellakin salasanoja.
- Palomuuriohjelmistot tai “rautapalomuurit”.
- Virusskanneri (etenkin Windowsissa).
- Käyttöjärjestelmää/ohjelmistoja pidetään ajan tasalla!
- Ei turhia palveluita!
- Ei epämääräisiä ohjelmia.
- Konetta ei käytetä jatkuvasti pääkäyttäjän oikeuksilla.
- Varotaan ns. sosiaalista hakkerointia!

## Tähtitieteilijöiden käyttämiä ohjelmistoja

Momcheva & Tollerud 2015 tekivät kyselyn tähtitieteilijöille heidän käyttämistään ohjelmistoista.

- 1142 vastausta.
- 90% vastaajista oli kirjoittanut ainakin osan käyttämistään ohjelmista itse.
- 8% oli saanut laajamittaista koulutusta ohjelmistokehityksestä.
- 43% ei ollut saanut mitään ohjelmistokoulutusta!
- Työkalujen Top 10 ovat: Python, shell-skriptit, IDL, C/C++, Fortran, IRAF, taulukkolaskenta, HTML/CSS, SQL ja – Supermongo!

## Matlab on #11!

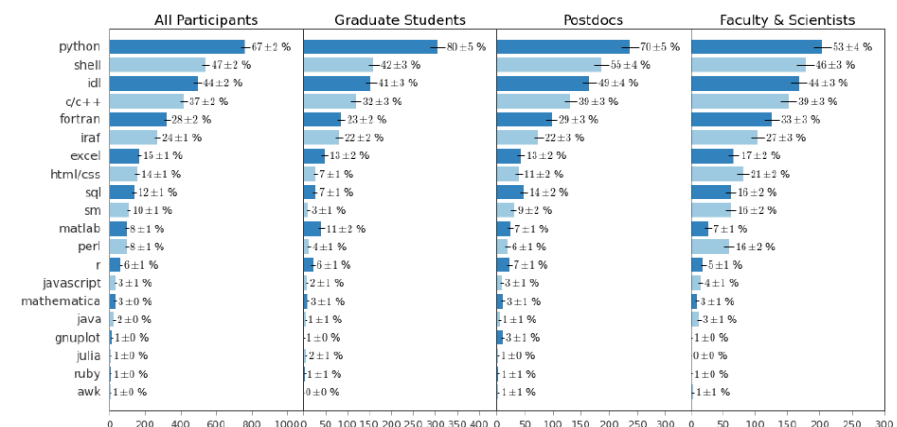


FIG. 10.— Responses to the prompt “Select any of these that you regularly use in your research”, sub-divided by career stage. The options listed included: IDL, IRAF, Python, C, Fortran, Perl, Javascript, Julia, Matlab, Java, R, SQL, Shell Scripting, STAN, Figaro, Ruby, HML/CSS, Supermongo (labeled “sm”), and Excel or other spreadsheets (labeled “excel”). Respondents could add additional tools not listed using an “Other” box. Among the tools in this plot, four items were added by respondents: C++, Mathematica, gnuplot and awk. Note that the x axis varies between panels.

## Joitakin kommentteja

Tulokset eivät riippuneet voimakkaasti uravaiheesta tai maantieteellisestä sijainnista – tähtitieteilijät näyttävät muodostavan maailmanlaajuisen yhteisön.

Monet listalla olevista työkaluista eivät ole varsinaisesti matemaattisia tai data-analyysissä käytettäviä ohjelmistoja, mutta ne ovat silti oleellinen osa esim. havaintodatan käsittelyprosessia.

Jotkin hyvin vanhat paketit kuten Supermongo voivat liittyä instrumenttispesifisiin ohjelmiin.

## Editorit

**Editorit:** ohjelmia joilla voi käsitellä ns. pelkkiä ASCII-tekstitiedostoja, esim. ohjelmalistauksia, skriptejä, asetustiedostoja.

Unix-maailmassa käytettyjä editoreja ovat esim. Emacs, pico, gedit ja vi.

Windows: esim. Notepad, Notepad++.

## Tekstinkäsittely

Tieteelliset artikkelit: LaTeX-koodia, joka kirjoitetaan jollakin editorilla.

Matemaattisen tekstin tuottaminen esim. MS Officen/LibreOfficen kaavaeditorilla on turhan työlästä – eikä jälkikään ole aina kovin kaunista.

Monet tieteelliset lehdet toivovat, että käsikirjoitukset tehdään käyttäen niiden omaa LaTeX-makropakettia.

LaTeX on saatavissa käytännössä kaikkiin käyttäjärjestelmiin.

LaTeX-koodi on puhdasta ASCII-tekstiä, jonka saa aina auki.

LateX:illa tehtyihin dokumentteihin voi liittää kuvia useissa formaateissa.

## Ohjelmointi

Tärkeimmät ohjelmointikielet tähtitieteessä ovat Fortran ja C/C++.

Ilmaisia kääntäjiä: gcc, gfortran.

Tulkkaavien kielten suosio on kasvussa: IDL, Python, Matlab ja Mathematica ovat näistä yleisimpiä.

Suurta numeronmurskaustehoa tarvittaessa turvaudutaan edelleen Fortraniin ja C:hen.

# Havaintojen käsittely ja esittäminen

Useita tähtitieteellisten havaintojen käsittelyyn suunniteltuja paketteja: IRAF, MIDAS, AIPS jne.

Tulosten visualisoinnissa IDL (Interactive Data Language) on tehokas yleistyökalu.

Kuvien käsittelyssä käytettäviä ohjelmia ovat myös Gimp ja Adobe Photoshop.

# Unix/Linux tunnissa

Unix on tietokoneen käyttöjärjestelmä, tarkemmin sanottuna monen käyttäjän moniajokäyttöjärjestelmä.

Tärkeimpiä kaupallisia Unixeja olivat AIX (IBM), HP-UX (HP), Solaris (Sun) jne.

Suluissa oleva kohta osoittaa, että ne olivat yleensä laitevalmistajan omia variantteja, olkoonkin että loppukäyttäjälle erot olivat yleensä pieniä.

Yhtenä Unix-varianttina voidaan pitää myös Linuxia, joka on avoimeen koodiin perustuva käyttöjärjestelmä joka on saatavissa useille laiteympäristöille.

Jos mennään halkomaan hiuksia, niin Linux on vain käyttöjärjestelmän ydin, suuri osa muista ohjelmista on peräisin esim. GNU-projektista. Joskus käytetäänkin nimitystä GNU/Linux.

Käytettävän käyttöjärjestelmäversion saa näkyviin komennolla:

```
uname -a
```

# Loggautuminen Unix/Linux-koneeseen

Login-prompti joko graafisena käyttöliittymänä tai komentorivillä:

**login:**

Tai

**Username:**

Käyttäjätunnuksen ilmoittamisen jälkeen kone kysyy salasanaa:

- mieluusti vähintään 8 merkkiä
- sekaisin isoja ja pieniä kirjaimia+numeroita
- ei sanakirjasta
- ei kuviota näppäimistöllä.

Vaikka Unixia voidaan käyttää pelkässä tekstipääteyhteydessä, niin nykyään useimmiten käytetään apuna ikkunointijärjestelmää.

Unixissa/Linuxissa ikkunointijärjestelmänä on yleensä X11 (yleensä puhutaan X:stä).

Karuhkon perus-X:än sijaan nykyään on yleensä X11:n päällä pyörivänä graafisena työpöytäympäristönä Gnome, KDE, Unity tai esim. XFCE.

Jos kone ei käynnistä automaattisesti graafista käyttöliittymää, niin tämän voi tehdä komennolla `startx`.

Koneesta poistuminen tapahtuu komennolla `exit` tai (tai tekstiyhteyden tapauksessa myös `logout`).

Jos käyttää graafista käyttöliittymää, niin `exit` komentoriviltä annettuina tarkoittaa vain kyseisestä terminaalista poistumista, tämän lisäksi täytyy loggautua erikseen pois valikosta käsin.

## Yhteydet muihin koneisiin

Yhteydet muihin Unix-koneisiin kannattaa tehdä käyttäen ssh-ohjelmaa, joka kryptaa käytettävän viestiliikenteen.

Jos käyttäjätunnus logattavassa koneessa on sama, niin voi kirjoittaa `ssh kone`.

Jos käyttäjätunnus on toinen, niin sitten voi loggautuakseen kirjoittaa `ssh kone -l tunnus`.

X-ikkunoinnin forwardointi ei ole välttämättä automaattisesti päällä → jos haluaa käyttää toisessa koneessa sijaitsevia graafisia sovelluksia, niin yhteys on otettava seuraavasti: `ssh -X kone`.

Unix/Linux-koneiden etäkäyttö onnistuu myös Windowsille tehdyistä ssh-toteutuksista. Graafisten ohjelmien käyttö onnistuu esim. Cygwinin xterminaalista käsin.

## Komentotulkki

Kun Unix-koneeseen ollaan loggauttu, niin käyttäjän kirjoittamia komentoja käsittelee ns. komentotulkki (engl. shell).

Yleisiä komentotulkkeja ovat bash, tcsh, csh ja zsh.

Näistä yleisin on nykyään bash, mutta tcsh on ainakin osalla oletuksena lastu-koneissa. Myös harjoituksissa käytettävä materiaali pohjautuu suurelta osin tcsh-tulkkiin.

Kun käyttäjä loggautuu koneeseen, käyttöjärjestelmä suorittaa joitakin alustustiedostoja.

tcsh:ssa näitä ovat mm. `.login` ja `.cshrc`, bashissa taas `.bashrc`.

Jos käyttäjällä ei ole omia versioitaan ko. tiedostosta, niin silloin käyttöön otetaan tiedostojen oletusversioita.

Määritellään esimerkkinä alias, eli lyhennysmerkintä vähän monimutkaisemmalle komennolle (tässä tapauksessa `ls`-komento kahdella parametrilla):

`.cshrc`:ssa tämän voisi tehdä rivillä

```
alias ll 'ls -al'
```

`.bashrc`:ssä taas rivillä

```
alias ll='ls -al'
```

`.cshrc` tai `.bashrc`-tiedostossa voi olla myös polkuasetuksia (ohjelmakirjastojen sijainti jne.).

Jos koneeseen on asennettu useampi shell, niin sitä voi tilapäisesti vaihtaa kirjoittamalla esim. bash:issä `tcsh`

# Hakemistorakenne ja tiedostot

Tieto sijaitsee kiintolevyllä olevissa tiedostoissa.

Käytön selkeyttämiseksi tiedostot on järjestetty hakemistorakenteeseen.

Levyllä on siis erilaisia hakemistoja, joissa voi olla tiedostoja ja edelleen alihakemistoja...

Komennolla `pwd` näkee oman sijaintinsa hakemistorakenteessa.

Listauksen hakemiston sisällöstä saa komennolla `ls`.

Pelkän `ls` -komennon antama informaatio on niukkaa, ja enemmän saa komennolla `ls -al`, missä komennolle `ls` annetaan argumentteina optiot `a` ja `l`.

Lisätietoa komennosta ja sen argumenteista saa kirjoittamalla `man komento`.

Uuden alihakemiston voi luoda komennolla `mkdir` olettaen, että käyttäjällä on kirjoitusoikeus kyseiseen paikkaan hakemistorakenteessa.

Tyhjän hakemiston voi tuhota komennolla `rmdir`.

Yksittäisen tiedoston voi tuhota komennolla `rm`, esimerkiksi `rm temppe`.

Hakemistorakenteessa on myös paikkoja, joihin tavallisella käyttäjällä ei ole kirjoittamisoikeutta.

Hakemistosta toiseen pääsee käyttämällä komentoa `cd`.

Kotihakemistoon pääsee pelkällä `cd` komennolla.

Hakemistorakenteen "juureen" pääsee komennolla `cd /`

## Muita tärkeitä komentoja

Tiedoston nimen voi muuttaa toiseksi käyttäen `mv`-komentoa:  
`mv tiedosto1 tiedosto2`

Tätä komentoa voi käyttää myös tiedoston siirtämiseen paikasta toiseen:

```
mv /hakemisto1/tiedosto /hakemisto2/tiedosto
```

Tiedoston voi kopioida toiselle nimelle `cp` komennolla:

```
cp tiedosto1 tiedosto2
```

Tämän jälkeen on olemassa kaksi sisällöltään identtistä tiedostoa.

Jos nimellä `tiedosto2` on jo ollut tiedosto, niin se tuhoutuu em. kopiointiprosessissa!

Tiedoston kopioimiseen koneesta toiseen käytetään yleensä `scp`-ohjelmaa:

```
scp tiedosto tunnus@kone:hakemisto  
-kopioidaan tiedosto toisen koneen hakemistoon.
```

```
scp tunnus@kone:hakemisto/tiedosto .  
-kopioidaan toisen koneen hakemistosta tiedosto siihen hakemistoon, jossa ollaan.
```

Kokonaisen hakemiston alihakemistoinen voi kopioida antamalla `scp`-komennolle argumenttin `-r`:

```
scp -r hakemisto tunnus@kone:.
```

Komennolla `cat` voi tulostaa (teksti)tiedoston sisällön:

```
cat tiedosto.
```

Koska tiedostot ovat usein pitkiä, edellinen komento ei ole aina kovin kätevä. Komento `more` toimii paremmin:

```
more tiedosto.
```

Tekstiä voi kelata joko ruutu (välilyönti) tai rivi (Enter) kerrallaan.

Moniin järjestelmiin on asennettu vieläkin kehittyneempi komento `less`, jolla tiedostoa voi selata molempiin suuntiin.

Kahden tiedoston väliset erot voi tulostaa komennolla `diff`:

```
diff tiedosto1 tiedosto2.
```

Haluttua merkkijonoa voi etsiä tiedostosta komennolla `grep`:

```
grep hassu tiedosto
```

→ tulostaa tiedostosta ne rivit, jotka sisältävät merkkijonon `hassu`.

Komento `sort` asettaa tiedoston riveittäin aakkosjärjestykseen (huom! katso man-sivulta komennon muita argumentteja).

Normaalisti edellisten komentojen tulostukset ohjautuvat ruudulle.

Ne voi myös ohjata tiedostoon käyttämällä `>` -merkkiä, esim.

```
sort tiedosto1 > tiedosto2
```

tulostaa tiedoston `tiedosto1` riveittäin aakkostettuna `tiedostoon2`.

Komentoja voi myös putkittaa `|`-merkkiä käyttäen:

```
grep hassu tiedosto |more
```

aiheuttaa sen, että tiedostosta “`tiedosto`” tulostetaan rivit, joilla esiintyy merkkijono “`hassu`”, tämä tulostus ohjataan edelleen `more` komennolle, jolloin sitä voidaan tarkastella rivi/ruutu kerrallaan.

Suuria tiedostoja, joita ei tarvita kovin usein voi kompressoida tiiviimpään muotoon.

```
compress tiedosto
```

kompressoi tiedoston sisällön ja luo tiiviimmän tiedoston nimellä `tiedosto.Z`.

Komennolla

```
uncompress tiedosto.Z
```

tiedosto palautuu alkuperäiseen muotoon.

Useissa järjestelmässä on myös pakkaussuhteeltaan tehokkaampi `gzip`, jonka pakkaus ja purkukomennot ovat `gzip` ja `gunzip`.

Myös `zip`-ohjelmisto on yleinen eri koneissa.

Järjestelmän yleisen levytilan käytön näkee komennolla `df`.

Yksittäisen hakemiston taas komennolla `du`.

Hakemistossa on usein suuri määrä tiedostoja. Niiden käsittelyssä on usein näppärää käyttää ns. wildcard-merkkejä.

```
cp tiedosto?.txt hakemisto2
```

kopioi kaikki tiedostot, joiden nimessä on sanan “`tiedosto`” jälkeen vielä yksi merkki ennen osiota “`.txt`” hakemistoon `hakemisto2`.

```
cp tiedosto*.txt hakemisto2
```

kopioi kaikki tiedostot, joiden nimessä on sanan `tiedosto` jälkeen jokin merkkijono ennen osiota “`.txt`” hakemistoon `hakemisto2`.



Usein on kätevää arkistoida useita tiedostoja tai jopa kokonaisia hakemistoja yksittäisiksi arkistotiedostoiksi.

```
tar cvf arkisto.tar *.jpg
```

arkistoi tiedostoon arkisto.tar kaikki tiedostot, joiden nimi päättyy merkkijonoon .jpg.

```
tar xvf arkisto.tar
```

puolestaan purkaa arkiston.

Huom! ennen purkamista kannattaa arkiston sisältö tarkistaa:

```
tar tvf arkisto.tar.
```

Myös zip-ohjelmistossa on mukana hakemistorakenteen pakkaaminen yhteen .zip-tiedostoon.

.zip- tai .gz-tiedoston sisällön voi tarkistaa myös graafisessa käyttöliittymässä purkamatta tiedostoa kiintolevyllä!

Yksittäisiä prosesseja voidaan käynnistää myös taustalle käyttämällä & -merkkiä, jolloin niiden käynnistämiseen käytetty terminaali pystyy edelleen vastaanottamaan komentoja, esim. `emacs tiedosto &`.

Jos haluaa muuttaa jo pyörivän prosessin taustaprosessiksi, niin se tapahtuu painamalla ensiksi prosessin käynnistäneessä xterminaalissa Ctrl-z ja kirjoittamalla sitten `bg`.

Käynnissä olevat prosessit näkee komennolla `ps` (huom! tarkista optiot `man-komennolla`).

Jos pyörivä prosessi, esim. jumiin jäänyt terminaali halutaan keskeyttää resurssien vapauttamiseksi, niin se tapahtuu komennolla `kill nro` tappaa prosessin, jonka numero `nro` on saatu komennolla `ps`.