

Lecture 2: computer use: Unix, compiling, and graphics

1. Essential unix commands
2. Emacs editor
3. Fortran compiler
4. Displaying graphics (gnuplot)

Essential unix commands

Logon/logoff procedure:

- Install Putty or Cygwin on your windows computer
- Putty: enter hostname (moe.cas.utk.edu or larry.cas.utk.edu) and click 'open'. Enter your password upon request.
- Cygwin: `ssh username@moe.cas.utk.edu`. Enter your password upon request.
- Now you have a 'shell' opened, and the command prompt is displayed.
- To exit: `exit` or `CRTL-d`

Essential unix commands

1. `pwd` print working directory
2. `cd` change directory
3. `ls` list (directory)
4. `cp` copy (file or directory)
5. `mv` move (file)
6. `rm` remove (file)
7. `mkdir` make directory (folder)
8. `mvdir` move directory
9. `rmdir` remove directory
10. `more` show content of file
11. `man` manual page (of command)
12. `ps` process status
13. `kill` kill (process #)
14. `CRTL-x` kills present process
15. `tcsh` more convenient shell (command completion/repetition)

Emacs editor

`emacs file &` opens new window with content of file 'file' for editing (Cygwin); returns command prompt

`emacs -nw file` no window option: opens file 'file' in present terminal window (Putty)

Commands within emacs

CTRL-k kills current line

CTRL-x CTRL-s saves file

CTRL-x CTRL-c exits emacs

copy and paste by highlighting text with left mouse button and pasting with middle mouse button.

Compiling fortran programs

<code>f90 prog2.f90</code>	compiles and links prog2.f90 → a.out
<code>f90 -o prog2.exe prog2.f90</code>	compiles and links prog2.f90 → prog2.exe
<code>f90 -c fun.f90</code>	compiles fun.f90 → fun.o
<code>f90 -c prog3.f90</code>	compiles prog3.f90 → prog3.o
<code>f90 -o prog3.x fun.o prog3.o</code>	links fun.o and prog3.o → prog3.x

Some compiler options

<code>-o name</code>	creates executable (“object”) file ‘name’
<code>-c</code>	compiles only (no linking)
<code>-g</code>	prepares for debugging (useful also w/o debugger, as details to crash are given); usually slow in execution
<code>-ftrap=all</code>	traps all floating point exceptions (underflow, overflow, division by zero)
<code>-O</code>	optimization
<code>-L /pathname/library</code>	linker links to library (with full pathname included)

gnuplot – simple graphics

gnuplot

plot 'fort.20'

plot "fort.20" w linesp

replot "fort.30"

set term dumb

set term post

set out 'plot.ps'

exit, quit, or q

starts gnuplot

plots data in file 'fort.20' as points

plots data in file "fort.20" with lines
points

replots previous plot + 'fort.30'

when display cannot be transported
(e.g. when using putty)

set terminal to postscript

will plot (output) to file "plot.ps"

quits gnuplot

graphics viewing and converting

`ghostview plot.ps`

view postscript file “plot.ps”

`ps2pdf plot.ps`

generates pdf-file plot.pdf from
postscript file plot.ps

`acroread plot.pdf`

view pdf file “plot.pdf”

Note: `acroread` and `ghostview` will not work with putty.

file transfer between computers

`scp file user@machine.utk.edu:./newfile`

securely copies file to newfile in home directory of user at machine.utk.edu

`scp user@machine.utk.edu:./file newfile`

securely copies file from user's home directory on machine.utk.edu to newfile in present directory

Use on unix machines or under cygwin.

On windows computer without cygwin

1. download pscp program
2. open windows command prompt
3. use pscp instead of scp. syntax as above. note: unix uses / in pathnames, while windows uses \