

Analysis Methods in Atmospheric and Oceanic Science

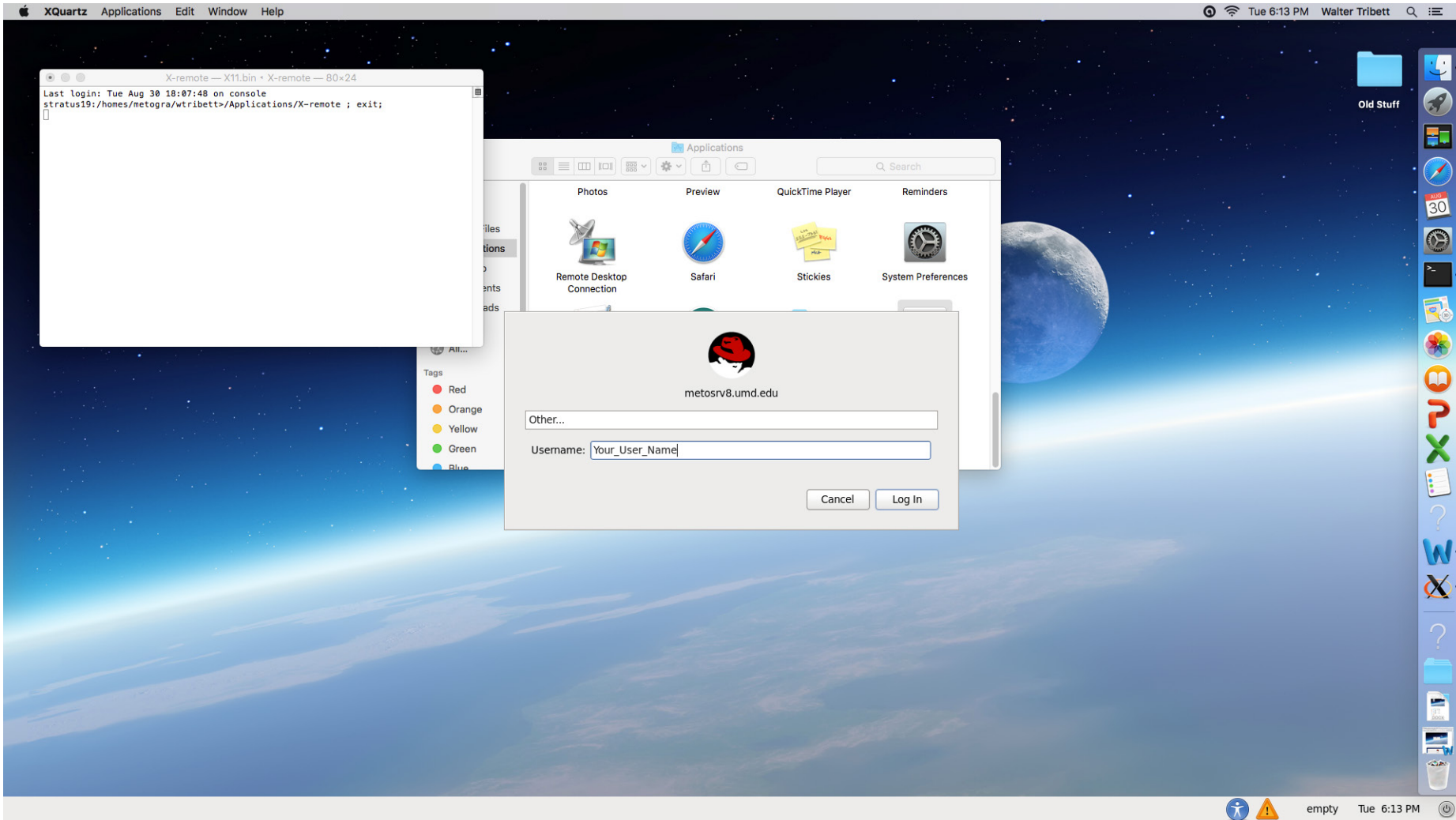
AOSC 652

Intro to FORTRAN and Simple Computations: Day 2

Web Site: <http://www.atmos.umd.edu/~rjs/class/fall2016/>

7 Sep 2016

Linux KDE Environment



For more info, see <http://www.kde.org> or <http://en.wikipedia.org/wiki/KDE>

AOSC 652: Student Editor Preferences

Prior Software	Number of Students
C or C++	5
FORTRAN	3
GrADS	3
IDL	0
Java	1
MATLAB	6
PYTHON	3
R	2
XLMA	1
WDSS-II	2

AOSC 652: Analysis Methods in AOSC

Steps to setup and use the Portland F77 compiler:

a) Be sure the following lines appear in your `.cshrc` file:

```
set path = ($path /usr/local/pgi-9.0.2/linux86-64/9.0-2/bin)
setenv PGI /usr/local/pgi-9.0.2
setenv LM_LICENSE_FILE $PGI/license.dat
```

b) Create file `.pgif77` in your home directory containing:

```
pgf77 -c -Msave -Mbounds -traceback $1.f
pgf77 -o $1.e $1.o
```

c) Add:

```
alias pf '~/pgif77'
```

to your `.aliases` file

d) `metosrv8{rjs} source .aliases`

e) `metosrv8{rjs} chmod u+x .pgif77`

f) `metosrv8{rjs} cp ~/rjs/aosc652/week_02/my_first_pgm.f .`

g) `metosrv8{rjs} pf my_first_pgm`

h) `metosrv8{rjs} my_first_pgm.e`

AOSC 652: File Permissions

```
% dir .aliases
```

```
12 -rw-r- -r- - 1 rjs aosc 9493 Sep 7 2014 .aliases
```

```
% dir .pgif77
```

```
4 -rwxr-xr-x 1 rjs aosc 60 Jun 15 2010 .pgif77*
```

See <https://www.linux.com/learn/understanding-linux-file-permissions>
for more info on file permissions

AOSC 652: File Permissions

```
% dir .aliases
```

```
12 -rw-r--r-- 1 rjs aosc 9493 Sep 7 2014 .aliases
```

6 4 4

```
% dir .pgif77
```

```
4 -rwxr-xr-x 1 rjs aosc 60 Jun 15 2010 .pgif77*
```

7 5 5

On Fri, I entered command:

```
% chmod u+x .pgif77
```

See <https://www.linux.com/learn/understanding-linux-file-permissions>
for more info on file permissions

AOSC 652: File Permissions

% dir .aliases

```
12 -rw-r--r-- 1 rjs aosc 9493 Sep 7 2014 .aliases
```

6 4 4

% dir .pgif77

```
4 -rwxr-xr-x 1 rjs aosc 60 Jun 15 2010 .pgif77*
```

7 5 5

On Fri, I entered command:

```
% chmod u+x .pgif77
```

See <https://www.linux.com/learn/understanding-linux-file-permissions>
for more info on file permissions

AOSC 652: File Permissions

```
% dir .aliases
```

```
12 -rw-r--r-- 1 rjs aosc 9493 Sep 7 2014 .aliases
```

6 4 4

```
% dir .pgif77
```

```
4 -rwxr-xr-x 1 rjs aosc 60 Jun 15 2010 .pgif77*
```

7 5 5

On Fri, I entered command:

```
% chmod u+x .pgif77
```

Command:

```
% chmod 744 .pgif77
```

would have had exact same effect

See <https://www.linux.com/learn/understanding-linux-file-permissions>
for more info on file permissions

FORTRAN Nomenclature

- Files whose names end with **.f** are considered to be FORTRAN source files
 - ASCII file containing FORTRAN commands written according to prescribed syntax
 - The actual FORTRAN code
- Files whose names end with **.o** are taken as object files, and are passed directly to the compiler if compiling is requested.
 - Contains “machine language”
 - See http://en.wikipedia.org/wiki/Object_code for more info
 - If you were a Computer Sciences major, you might be exposed to machine language: not for the “faint of heart”
- Common convention: files whose names end with **.e** are “executable” files
 - Multiple **.o** files can be compiled into a single **.e** executable
 - The actual FORTRAN program
 - See <http://en.wikipedia.org/wiki/Executable> for more info

FORTRAN Precision

Floating-point

Default is single precision: 32 bits or 4 bytes

FORTRAN default is:

implicit single precision (a-h, o-z) or
implicit real*4 (a-h,o-z)

Can use double precision: 64 bits or 8 bytes

implicit double precision (a-h, o-z) or
implicit real*8 (a-h, o-z)

Note: the statements

```
double precision x  
x = 0.0025
```

will lead to x being represented as something other than
2.5000000000000000E-02 on many compilers

AOSC 652: Analysis Methods in AOSC

Copy file ~rjs/aosc652/week_02/precision1.f
~rjs/aosc652/week_02/precision2.f

to your work area.

AOSC 652: Analysis Methods in AOSC

Copy file `~rjs/aosc652/week_02/precision1.f`
`~rjs/aosc652/week_02/precision2.f`

to your work area.

Compile `precision1.f`

AOSC 652: Analysis Methods in AOSC

Copy file `~rjs/aosc652/week_02/precision1.f`
`~rjs/aosc652/week_02/precision2.f`

to your work area.

Compile `precision1.f`

Run `precision1.e`

FORTRAN Precision

Floating-point

Default is single precision: 32 bits or 4 bytes

FORTRAN default is:

implicit single precision (a-h, o-z) or
implicit real*4 (a-h,o-z)

Can use double precision: 64 bits or 8 bytes

implicit double precision (a-h, o-z) or
implicit real*8 (a-h, o-z)

Note: the statements

double precision x
x = 0.0025

will lead to x being represented as something other than
2.5000000000000000E-02 on many compilers

**How do we need to alter the statement “x = ...” for x to be
evaluated as 2.5000000000000000E-02 ?**

AOSC 652: Analysis Methods in AOSC

Copy file `~rjs/aosc652/week_02/precision1.f`
`~rjs/aosc652/week_02/precision2.f`

to your work area.

Compile & run `precision2.f`

FORTRAN Format Statement

				<u>Example</u>
Integer	I	I4	4 positions reserved	2008
Floating-point				
	F	F7.2	Decimal notation 7 positions, 2 of which come after decimal point	-654.21
	E	1PE9.2	Scientific notation 9 positions: 1 before decimal point, 2 after decimal point	-8.65E-21

Note: if 1P is used in a format statement, and an Fw.d command follows in the same format statement, will get erroneous output unless 0PFw.d is used

Character	A	A8	8 positions	AOSC 652
-----------	---	----	-------------	----------

For more info, see:

<http://www.cs.mtu.edu/~shene/COURSES/cs201/NOTES/chap05/format.html>
as well as chapter 9 of Ellis

AOSC 652: Analysis Methods in AOSC

Copy file `~rjs/aosc652/week_02/format_example.f`

to your work area.

AOSC 652: Analysis Methods in AOSC

Copy file `~rjs/aosc652/week_02/format_example.f`

to your work area.

Compile and run

AOSC 652: Analysis Methods in AOSC

Format statements:

In general, we prefer *unformatted reads* and formatted writes:

```
program format_test
character*6 cdate

write(6,700)
700 format('Enter date : ', $)
read(5,*)cdate ←
write(6,702)cdate
702 format('cdate =', A6)

write(6,704)
704 format('Enter latitude (deg, min, sec): ', $)
read(5,*)xlat_deg,xlat_min,xlat_sec

write(6,706)xlat_deg,xlat_min,xlat_sec
706 format('Lat =', F5.0, ' deg, ', F5.0, ' min, ', F7.2, ' sec')

stop
end
```

AOSC 652: Analysis Methods in AOSC

Format statements:

In general, we prefer unformatted reads and *formatted writes*:

```
program format_test
character*6 cdate

write(6,700)
700 format('Enter date : ', $)
read(5,*)cdate

write(6,702)cdate ←
702 format('cdate =', A6)

write(6,704)
704 format('Enter latitude (deg, min, sec): ', $)
read(5,*)xlat_deg,xlat_min,xlat_sec

write(6,706)xlat_deg,xlat_min,xlat_sec
706 format('Lat =', F5.0, ' deg, ', F5.0, ' min, ', F7.2, ' sec')

stop
end
```

AOSC 652: Analysis Methods in AOSC

Copy files ~rjs/aosc652/week_02/trig1.f
~rjs/aosc652/week_02/trig2.f
~rjs/aosc652/week_02/trig3.f
~rjs/aosc652/week_02/trig4.f

to your work area.

AOSC 652: Analysis Methods in AOSC

4, 3

```
degree, cos, cos^2+sin^2, 1.E8*(cos^2+sin^2-1.)
```

Output of program trig1.f

0.0	1.00000E+00	1.00000E+00	0.00000E+00
1.0	9.9985E-01	1.00000E+00	-5.9605E+00
2.0	9.9939E-01	1.00000E+00	0.00000E+00
3.0	9.9863E-01	1.00000E+00	0.00000E+00
4.0	9.9757E-01	1.00000E+00	0.00000E+00
5.0	9.9620E-01	1.00000E+00	-5.9605E+00
6.0	9.9453E-01	1.00000E+00	0.00000E+00

AOSC 652: Analysis Methods in AOSC

of columns of data

4, 3

```
degree, cos, cos^2+sin^2, 1.E8*(cos^2+sin^2-1.)
```

```
Output of program trig1.f
```

0.0	1.00000E+00	1.00000E+00	0.00000E+00
1.0	9.9985E-01	1.00000E+00	-5.9605E+00
2.0	9.9939E-01	1.00000E+00	0.00000E+00
3.0	9.9863E-01	1.00000E+00	0.00000E+00
4.0	9.9757E-01	1.00000E+00	0.00000E+00
5.0	9.9620E-01	1.00000E+00	-5.9605E+00
6.0	9.9453E-01	1.00000E+00	0.00000E+00

AOSC 652: Analysis Methods in AOSC

of header lines

4, 3

```
degree, cos, cos^2+sin^2, 1.E8*(cos^2+sin^2-1.)
```

```
Output of program trig1.f
```

0.0	1.00000E+00	1.00000E+00	0.00000E+00
1.0	9.9985E-01	1.00000E+00	-5.9605E+00
2.0	9.9939E-01	1.00000E+00	0.00000E+00
3.0	9.9863E-01	1.00000E+00	0.00000E+00
4.0	9.9757E-01	1.00000E+00	0.00000E+00
5.0	9.9620E-01	1.00000E+00	-5.9605E+00
6.0	9.9453E-01	1.00000E+00	0.00000E+00

AOSC 652: Analysis Methods in AOSC

**Brief words describing content of each column,
separated by comma**

4, 3

degree, cos, cos²+sin², 1.E8*(cos²+sin²-1.)

Output of program trig1.f

0.0	1.00000E+00	1.00000E+00	0.00000E+00
1.0	9.9985E-01	1.00000E+00	-5.9605E+00
2.0	9.9939E-01	1.00000E+00	0.00000E+00
3.0	9.9863E-01	1.00000E+00	0.00000E+00
4.0	9.9757E-01	1.00000E+00	0.00000E+00
5.0	9.9620E-01	1.00000E+00	-5.9605E+00
6.0	9.9453E-01	1.00000E+00	0.00000E+00

AOSC 652: Analysis Methods in AOSC

4,3

```
degree,cos,cos^2+sin^2,1.E8*(cos^2+sin^2-1.)
```

Output of program trig1.f ← **Comment**

0.0	1.00000E+00	1.00000E+00	0.00000E+00
1.0	9.9985E-01	1.00000E+00	-5.9605E+00
2.0	9.9939E-01	1.00000E+00	0.00000E+00
3.0	9.9863E-01	1.00000E+00	0.00000E+00
4.0	9.9757E-01	1.00000E+00	0.00000E+00
5.0	9.9620E-01	1.00000E+00	-5.9605E+00
6.0	9.9453E-01	1.00000E+00	0.00000E+00

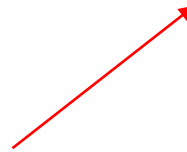
AOSC 652: FORTRAN Example

- Webpage for this week:

Useful links:

Introduction to FORTRAN written for AOSC students, by an AOSC student:

<http://www.atmos.umd.edu/~gcm/usefuldocs/fortran.html>



How many students explored this link ?

AOSC 652: FORTRAN Example

- Webpage for this week:

A Fortran 77 Demo

- Open the demo code at:
`/~scchan/tutorial/learn_f.f`
- Copy code to your home directory
- Open the code with emacs (or vi if you know how to use it)
 - `emacs learn_f.f`
 - `vi learn_f.f`

AOSC 652: FORTRAN Example

- Webpage for this week:

A Fortran 77 Demo

Anyone run this code ?

- Open the demo code at:
`/~scchan/tutorial/learn_f.f`
- Copy code to your home directory
- Open the code with emacs (or vi if you know how to use it)
 - `emacs learn_f.f`
 - `vi learn_f.f`

AOSC 652: FORTRAN Example

- Webpage for this week:

A Fortran 77 Demo

Code is located in `~rjs/aosc652/week_02/sschan_code`

- Open the demo code at:
`/~scchan/tutorial/learn_f.f`
- Copy code to your home directory
- Open the code with emacs (or vi if you know how to use it)
 - `emacs learn_f.f`
 - `vi learn_f.f`

AOSC: HW #2

We have given you the tools (readings, handouts/examples) to complete this assignment

Your decision how to enter your selected locations ... you could for example
“hard wire” these numbers into the code (brute force method)
or you can allow user input (more elegant and versatile)

Your decision ... our focus is **GETTING THE COMPUTATIONAL JOB DONE**
one way or another 😊

AOSC: HW #2

We have given you the tools (readings, handouts/examples) to complete this assignment

Your decision how to enter your selected locations ... you could for example “hard wire” these numbers into the code (brute force method)
or you can allow user input (more elegant and versatile)

Your decision ... our focus is **GETTING THE COMPUTATIONAL JOB DONE**
one way or another 😊 ⇒ **correctly, of course !**

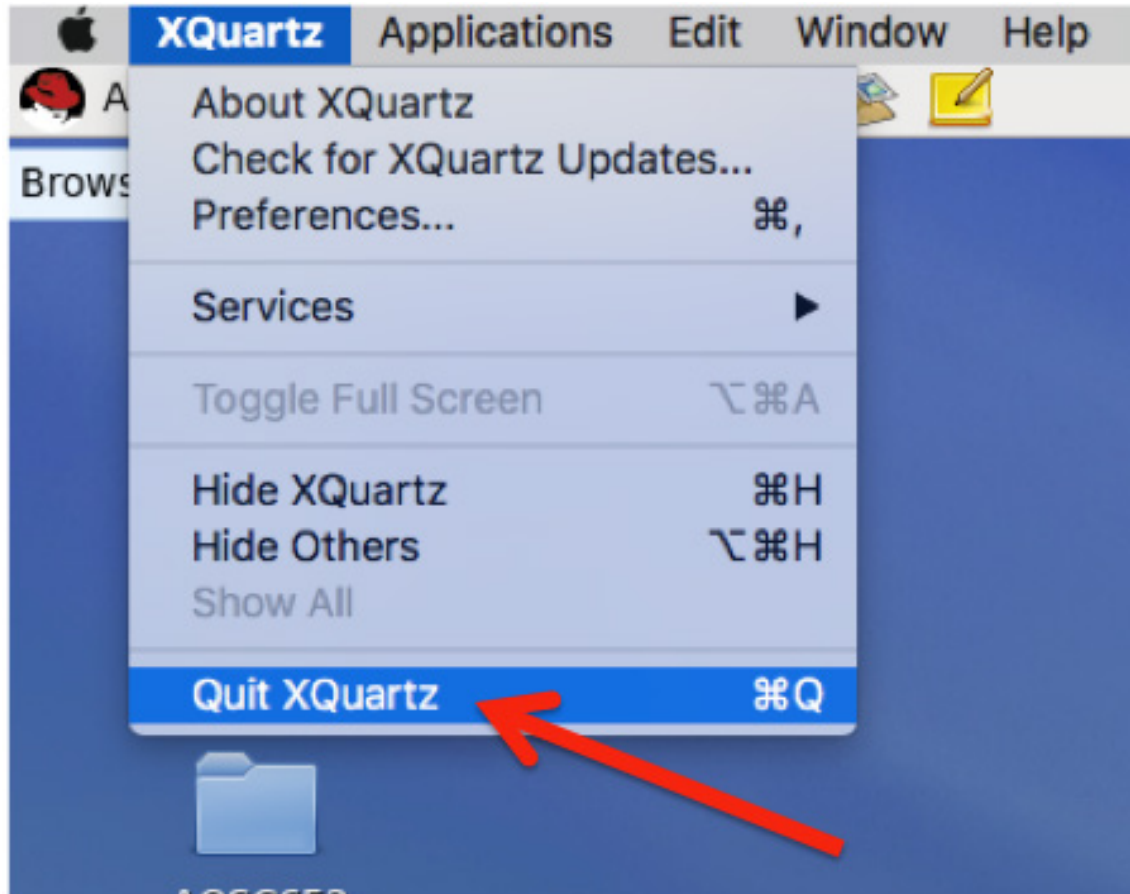
Friday will be devoted to in class work on Hw #2

- Exercise involves use of [Google Earth](#):
 - Google Earth application is available on computers in this room (Mac environment)
 - ***If you'd can complete the Google Earth portion prior to class on Friday, this will be appreciated 😊***
 - *Please have a look at the assignment prior to class; you are also welcome to start working on the entire assignment prior to class !*
- Assignment #2 will be the **last one** that we think can perhaps be completed during the ~1 hr class session (nonetheless, not due until Monday, 12 Sept)
- For efficient use of Friday portion of class **[please begin Assignment prior to class !](#)**

AOSC 652: Logging out Properly Is Important

Log out Steps

1. To log out click the XQuartz on the top menu bar and select “Quit XQuartz”. (If you do not see XQuartz at the top try clicking the Linux wallpaper background on the screen)



AOSC 652: Logging out Properly Is Important

Log out Steps

2. Click the apple in the top left corner and select “Log Out”

