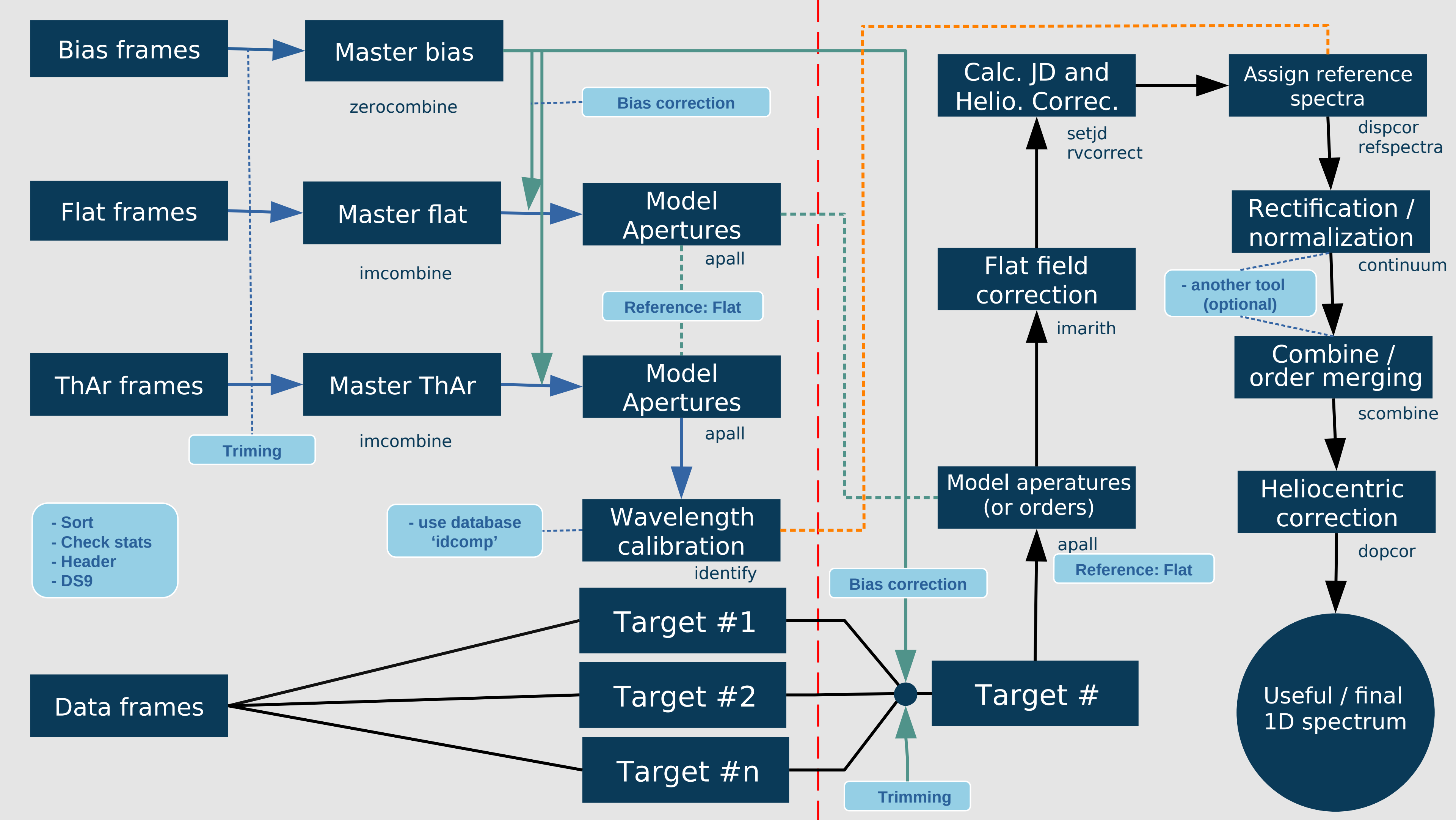
Data reduction of the Echelle spectra using IRAF

Mauricio Cabezas, Matti Dorsch

29.08.2024

Workshop on observational techniques

26 August – 6 September 2024 at Ondřejov observatory



Bias frames

Master bias

zerocombine

Bias correction

Flat frames

Master flat

imcombine

Model Apertures

apall

Reference: Flat

ThAr frames

Master ThAr

imcombine

Model Apertures

apall

Triming

- Sort
- Check stats
- Header
- DS9

- use database ‘idcomp’

Wavelength calibration

identify

Data frames

Target #1

Target #2

Target #n

Bias correction

Trimming

Target #

Model aperatures (or orders)

apall

Reference: Flat

Flat field correction

imarith

Calc. JD and Helio. Correc.

setjd
rvcorrect

Assign reference spectra

dispcor
refspectra

Rectification / normalization

continuum

- another tool (optional)

Combine / order merging

scombine

Heliocentric correction

dopcor

Useful / final 1D spectrum

Set up your folders

Open a terminal

You should probably put the raw data in the “Share” folder (might run out of space after a few days of data)

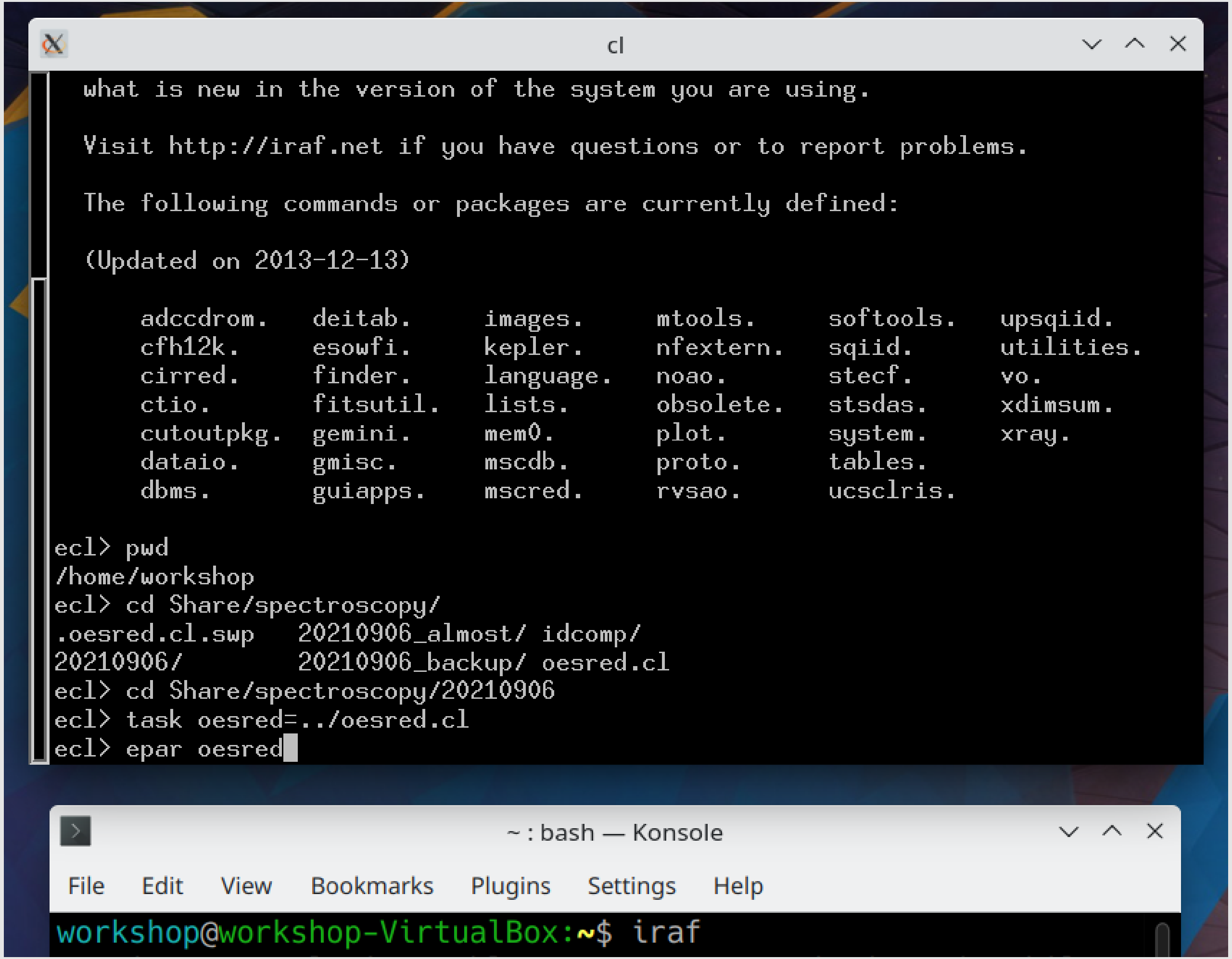
```
mkdir  ~/Share/spectroscopy
```

```
cp -r 20240828
~/Share/spectroscopy
```

Keep a copy of the original files!

Get the reduction script

```
cp -r
~/spectroscopy_project/*
~/Share/spectroscopy
```



```
  what is new in the version of the system you are using.

  Visit http://iraf.net if you have questions or to report problems.

  The following commands or packages are currently defined:

  (Updated on 2013-12-13)

      adccdrom.   deitab.     images.     mtools.     softools.   upsqiid.
      cfh12k.     esowfi.     kepler.     nfextern.   sqiid.      utilities.
      cirred.     finder.     language.   noao.       stecf.      vo.
      ctio.       fitsutil.   lists.      obsolete.   stsdas.     xdimsum.
      cutoutpkg.  gemini.     mem0.       plot.       system.     xray.
      dataio.     gmisc.      mscdb.      proto.      tables.
      dbms.       guiapps.    mscred.     rvsao.      ucsclris.

ecl> pwd
/home/workshop
ecl> cd Share/spectroscopy/
.oesred.cl.swp   20210906_almost/ idcomp/
20210906/        20210906_backup/ oesred.cl
ecl> cd Share/spectroscopy/20210906
ecl> task oesred=../oesred.cl
ecl> epar oesred
```

```
workshop@workshop-VirtualBox:~$ iraf
```

Start IRAF

Open a terminal

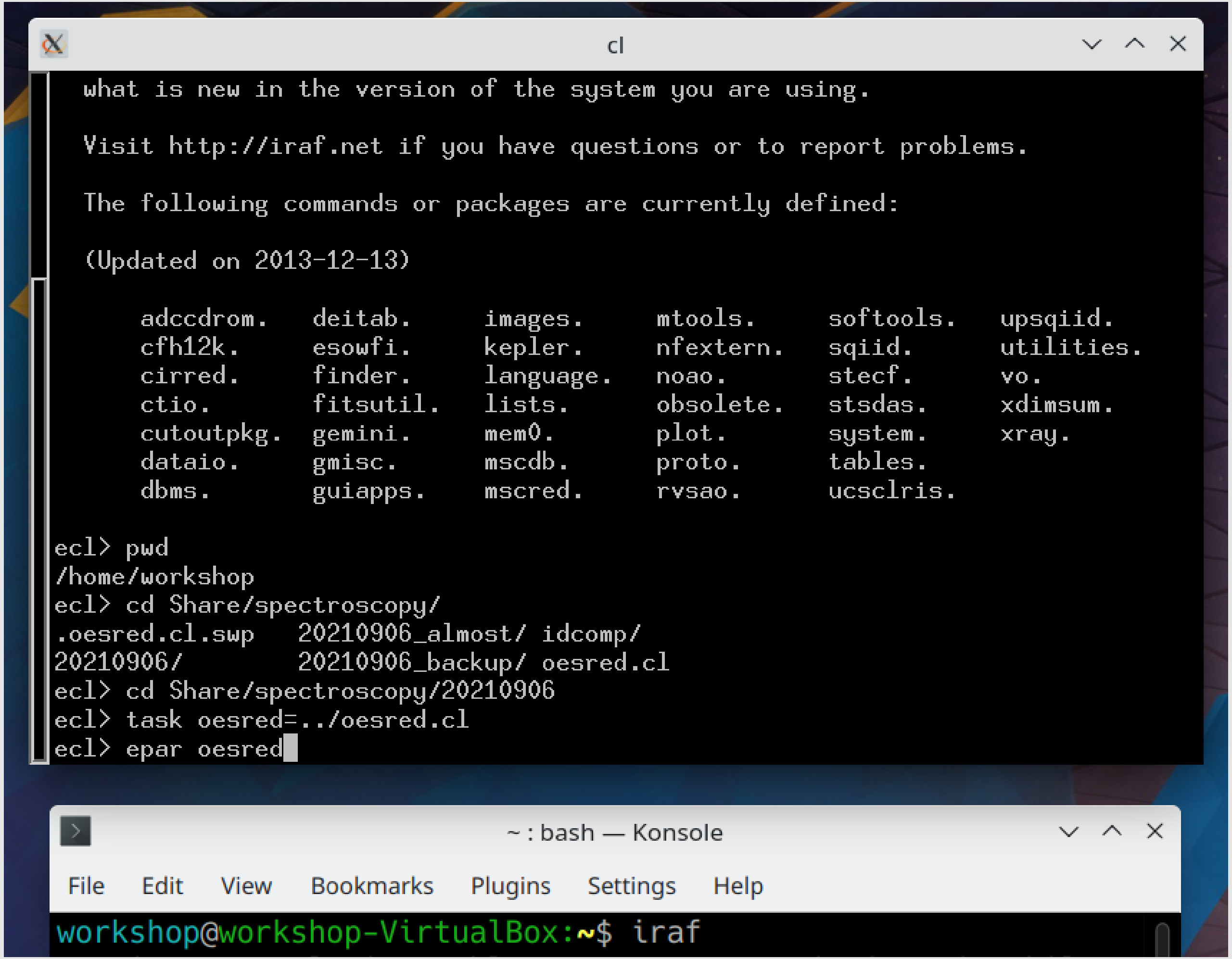
In the **home** directory:

```
iraf
```

Then navigate to the folder with the raw data

To look at images:

```
!ds9 &
display e20230629011.fit 1
```



```
  what is new in the version of the system you are using.

  Visit http://iraf.net if you have questions or to report problems.

  The following commands or packages are currently defined:

  (Updated on 2013-12-13)

      adccdrom.   deitab.     images.     mtools.     softools.   upsqiid.
      cfh12k.     esowfi.     kepler.     nfextern.   sqiid.      utilities.
      cirred.     finder.     language.   noao.       stecf.      vo.
      ctio.       fitsutil.   lists.      obsolete.   stsdas.     xdimsum.
      cutoutpkg.  gemini.     mem0.       plot.       system.     xray.
      dataio.     gmisc.      mscdb.      proto.      tables.
      dbms.       guiapps.    mscred.     rvsao.      ucsclris.

ecl> pwd
/home/workshop
ecl> cd Share/spectroscopy/
.oesred.cl.swp   20210906_almost/ idcomp/
20210906/        20210906_backup/ oesred.cl
ecl> cd Share/spectroscopy/20210906
ecl> task oesred=../oesred.cl
ecl> epar oesred
```

```
workshop@workshop-VirtualBox:~$ iraf
```

OESRED.CL

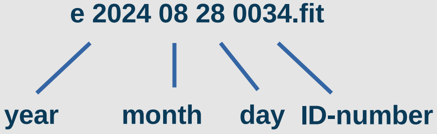
- SEMI automatic

- Parameters were tested and work exclusively for OES.

- Divided in two parts: Calibration and Science.
  Do the calibration part once per night

- You folder structure should look similar to this:

- **~/Share/spectroscopy/**
    - **20240828**
        - **e2024*.fit**
    - **20240830**
        - **e2024*.fit**
    - **idcomp**
    - **oesred.cl**



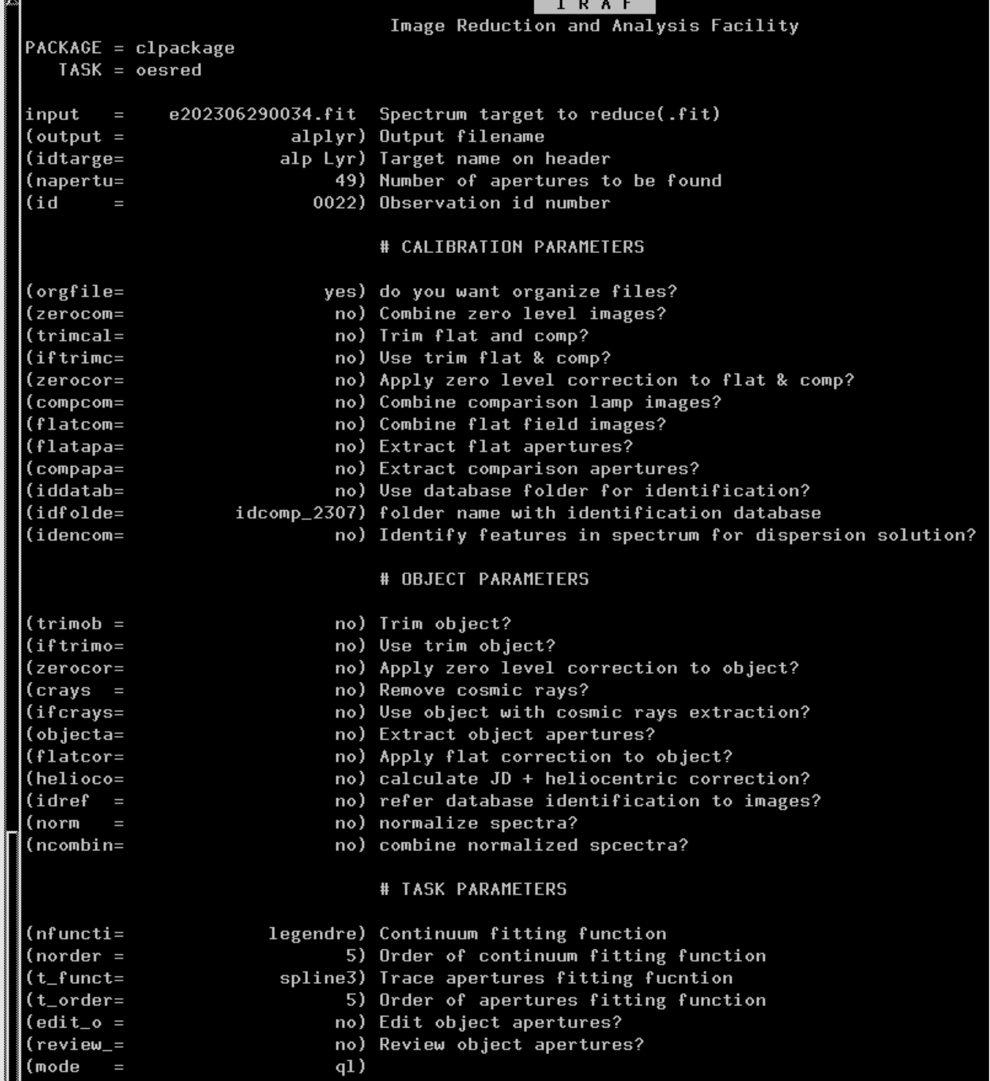
e 2024 08 28 0034.fit

year **month** **day** **ID-number**

task oesred = ../oesred.cl

***epar oesred** (or any task- **e**dit **par**ameter)*

*Quit **:q** Go! **:g** (do settings first)*



```
                                          I R A F
                                Image Reduction and Analysis Facility
PACKAGE = clpackage
   TASK = oesred

input   =     e202306290034.fit  Spectrum target to reduce(.fit)
(output =                alplyr) Output filename
(idtarge=               alp Lyr) Target name on header
(napertu=                    49) Number of apertures to be found
(id     =                  0022) Observation id number

                                 # CALIBRATION PARAMETERS

(orgfile=                   yes) do you want organize files?
(zerocom=                    no) Combine zero level images?
(trimcal=                    no) Trim flat and comp?
(iftrimc=                    no) Use trim flat & comp?
(zerocor=                    no) Apply zero level correction to flat & comp?
(compcom=                    no) Combine comparison lamp images?
(flatcom=                    no) Combine flat field images?
(flatapa=                    no) Extract flat apertures?
(compapa=                    no) Extract comparison apertures?
(iddatab=                    no) Use database folder for identification?
(idfolde=           idcomp_2307) folder name with identification database
(idencom=                    no) Identify features in spectrum for dispersion solution?

                                 # OBJECT PARAMETERS

(trimob =                    no) Trim object?
(iftrimo=                    no) Use trim object?
(zerocor=                    no) Apply zero level correction to object?
(crays  =                    no) Remove cosmic rays?
(ifcrays=                    no) Use object with cosmic rays extraction?
(objecta=                    no) Extract object apertures?
(flatcor=                    no) Apply flat correction to object?
(helioco=                    no) calculate JD + heliocentric correction?
(idref  =                    no) refer database identification to images?
(norm   =                    no) normalize spectra?
(ncombin=                    no) combine normalized spcectra?

                                 # TASK PARAMETERS

(nfuncti=              legendre) Continuum fitting function
(norder =                     5) Order of continuum fitting function
(t_funct=               spline3) Trace apertures fitting fucntion
(t_order=                     5) Order of apertures fitting function
(edit_o =                    no) Edit object apertures?
(review_=                    no) Review object apertures?
(mode   =                    ql)
```

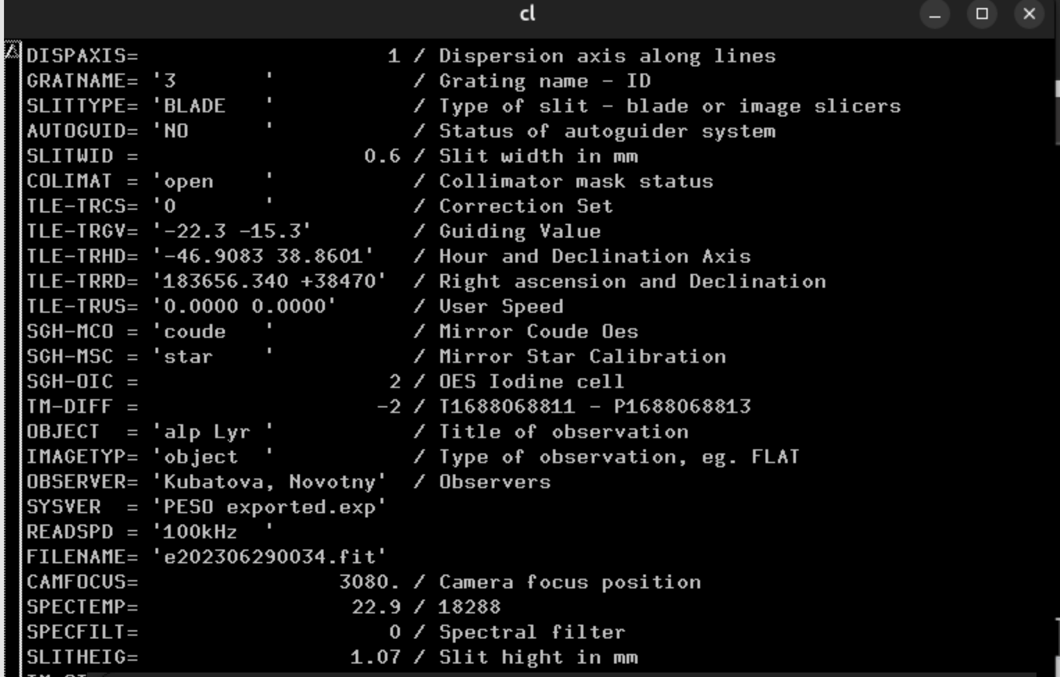
1) Initial parameters

Check type of “image”:

flat: flat fields
zero: bias
comp: ThAr comparison spectra
object: science

- if you want to see the full header

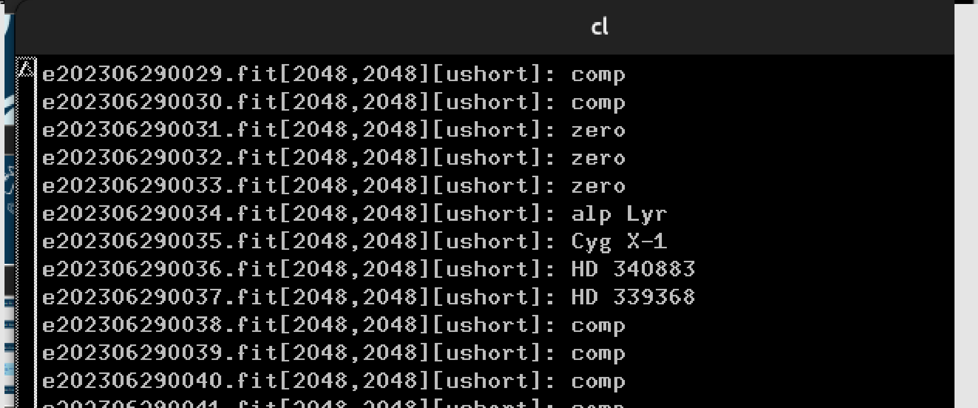
```
imhead e202109060001.fit l+ | page
```



```
DISPAXIS=                    1 / Dispersion axis along lines
GRATNAME= '3       '           / Grating name - ID
SLITTYPE= 'BLADE   '           / Type of slit - blade or image slicers
AUTOGUID= 'NO      '           / Status of autoguider system
SLITWID =                  0.6 / Slit width in mm
COLIMAT = 'open    '           / Collimator mask status
TLE-TRCS= '0       '           / Correction Set
TLE-TRGV= '-22.3 -15.3'        / Guiding Value
TLE-TRHD= '-46.9083 38.8601'   / Hour and Declination Axis
TLE-TRRD= '183656.340 +38470'  / Right ascension and Declination
TLE-TRUS= '0.0000 0.0000'      / User Speed
SGH-MCO = 'coude   '           / Mirror Coude Oes
SGH-MSC = 'star    '           / Mirror Star Calibration
SGH-OIC =                    2 / OES Iodine cell
TM-DIFF =                   -2 / T1688068811 - P1688068813
OBJECT  = 'alp Lyr '           / Title of observation
IMAGETYP= 'object  '           / Type of observation, eg. FLAT
OBSERVER= 'Kubatova, Novotny'  / Observers
SYSVER  = 'PESO exported.exp'
READSPD = '100kHz  '
FILENAME= 'e202306290034.fit'
CAMFOCUS=                3080. / Camera focus position
SPECTEMP=                 22.9 / 18288
SPECFILT=                    0 / Spectral filter
SLITHEIG=                 1.07 / Slit hight in mm
```

- print list filename and object type

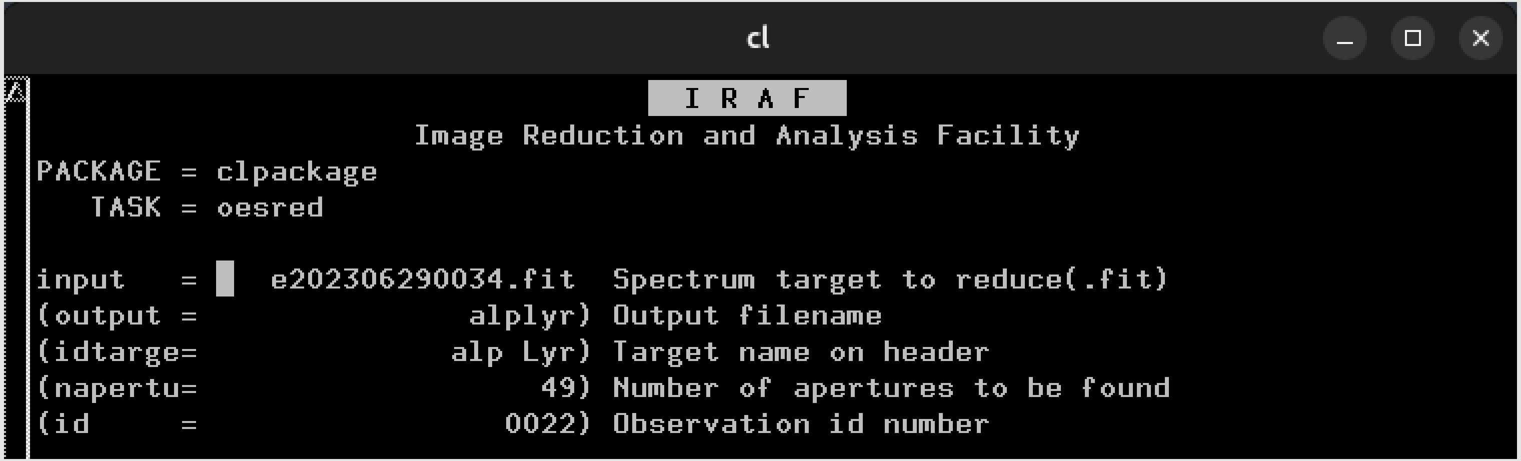
```
imhead e*.fit
```



```
e202306290029.fit[2048,2048][ushort]: comp
e202306290030.fit[2048,2048][ushort]: comp
e202306290031.fit[2048,2048][ushort]: zero
e202306290032.fit[2048,2048][ushort]: zero
e202306290033.fit[2048,2048][ushort]: zero
e202306290034.fit[2048,2048][ushort]: alp Lyr
e202306290035.fit[2048,2048][ushort]: Cyg X-1
e202306290036.fit[2048,2048][ushort]: HD 340883
e202306290037.fit[2048,2048][ushort]: HD 339368
e202306290038.fit[2048,2048][ushort]: comp
e202306290039.fit[2048,2048][ushort]: comp
e202306290040.fit[2048,2048][ushort]: comp
e202306290041.fit[2048,2048][ushort]: comp
```

- for example, alp Lyr spectrum.

- take note about the filename and the target name in the header!



```
                                  I R A F
                    Image Reduction and Analysis Facility
PACKAGE = clpackage
   TASK = oesred

input   =    e202306290034.fit  Spectrum target to reduce(.fit)
(output =               alplyr) Output filename
(idtarge=              alp Lyr) Target name on header
(napertu=                   49) Number of apertures to be found
(id     =                 0022) Observation id number
```

input: The complete fit file name of our target, science → see imhead, either type or paste with “Shift + Insert” shortcut

output: If output = “alplyr”, the filename of the final reduced spectrum will be “DCN-alplyr_20230929.fit” at the end of the reduction

idtarget: The name of the target to be written in the header

naperture: Number of apertures, important if we want to use the wavelength calibration database. Keep at 49.

2) sort files!

- complete first set of parameters, in the part of calibration set:

orgfile = yes

:go

After this step files are organized, you can check each folder and files.

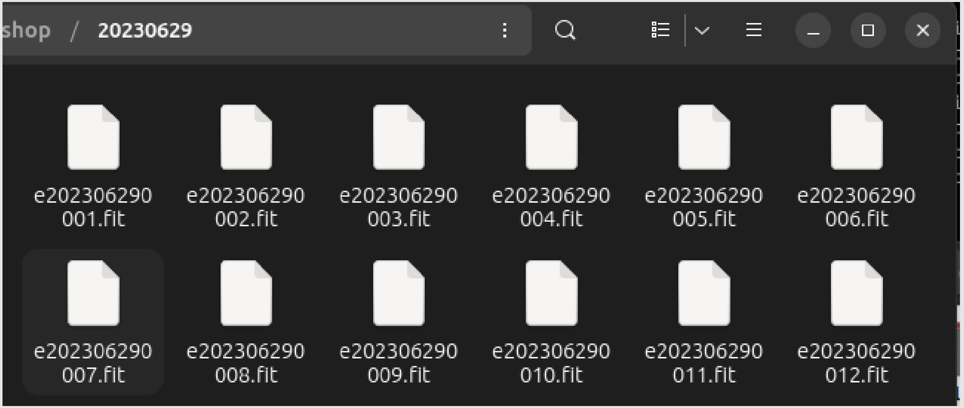
Example:

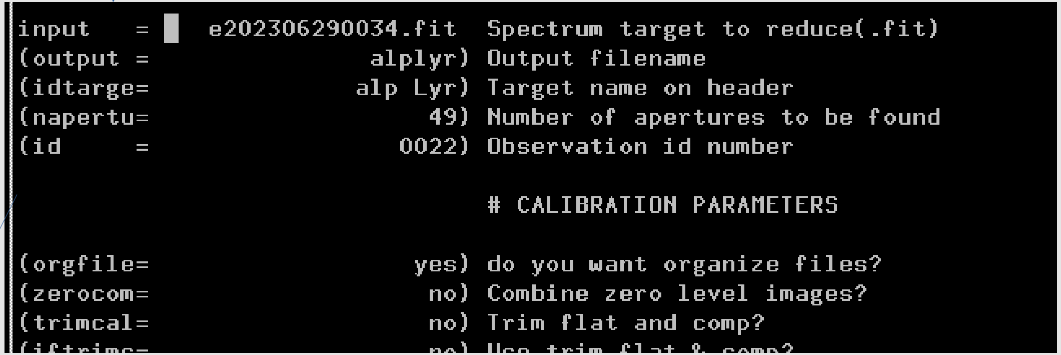
imstat @flat.dat

```
#           IMAGE        NPIX      MEAN    STDDEV     MIN      MAX
  e202109060006.fit   4194304    931.1    2621.      0.    25203.
  e202109060007.fit   4194304    949.1    2679.      0.    25797.
  e202109060008.fit   4194304    944.7    2664.      0.    25560.
  e202109060009.fit   4194304    941.4    2654.      0.    25521.
  e202109060010.fit   4194304    940.2    2649.      0.    25405.
```

- visual inspection:

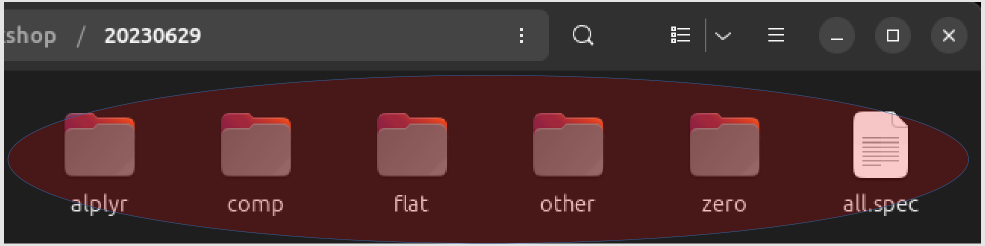
!ds9 &
display e202306290011.fit 1





```
input    =    e202306290034.fit  Spectrum target to reduce(.fit)
(output  =               alplyr) Output filename
(idtarge=               alp Lyr) Target name on header
(napertu=                    49) Number of apertures to be found
(id      =                 0022) Observation id number

                                 # CALIBRATION PARAMETERS

(orgfile=                   yes) do you want organize files?
(zerocom=                    no) Combine zero level images?
(trimcal=                    no) Trim flat and comp?
(iftrims=                    no) Use trim flat & comp?
```

Before each run be sure we are in the MAIN folder (20230629/)!
pwd



3) Combine zeros!

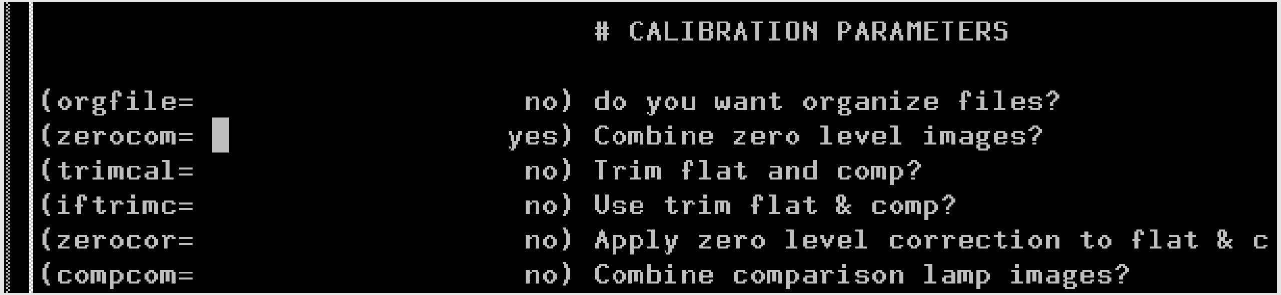
- Bias frames are collected in **zero.dat**
- change the previous task to “no”

  orgfile = no

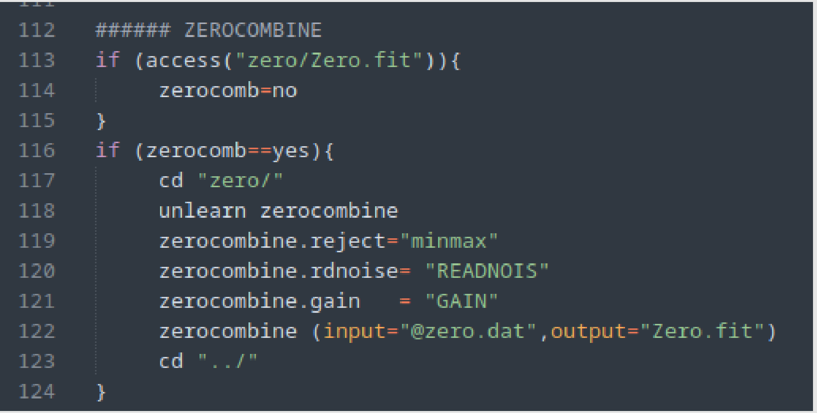
  zerocom = yes

  :go

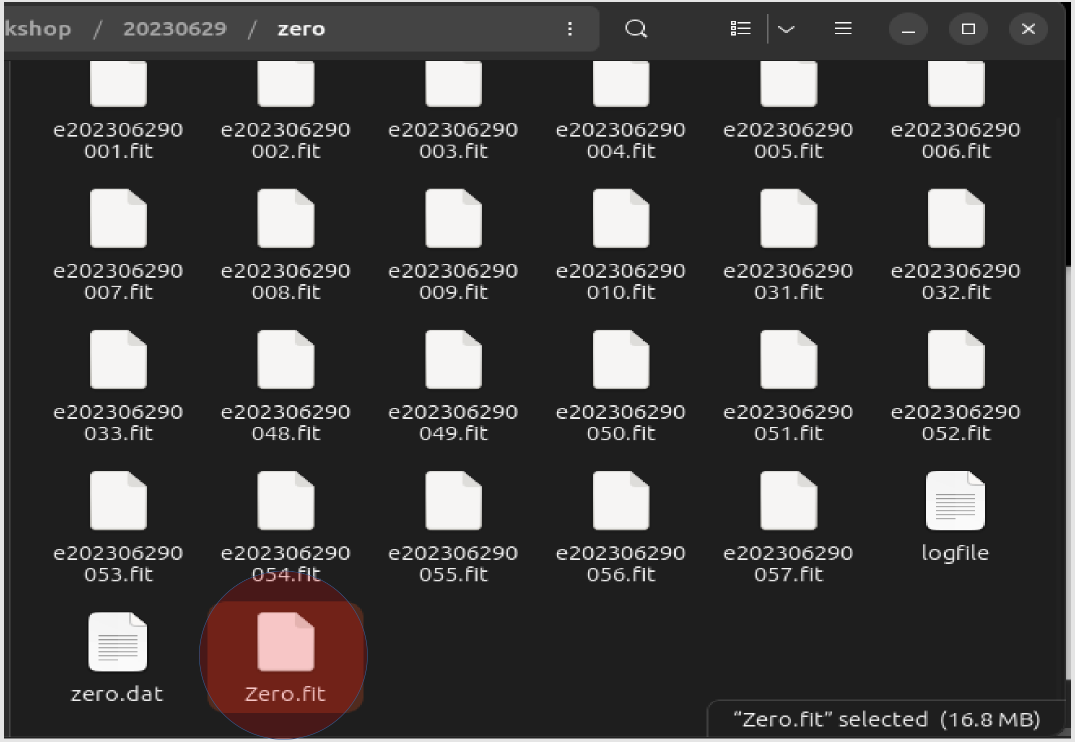
- new file: **Zero.fit**



```
                                    # CALIBRATION PARAMETERS

(orgfile=                    no) do you want organize files?
(zerocom=                   yes) Combine zero level images?
(trimcal=                    no) Trim flat and comp?
(iftrimc=                    no) Use trim flat & comp?
(zerocor=                    no) Apply zero level correction to flat & c
(compcom=                    no) Combine comparison lamp images?
```



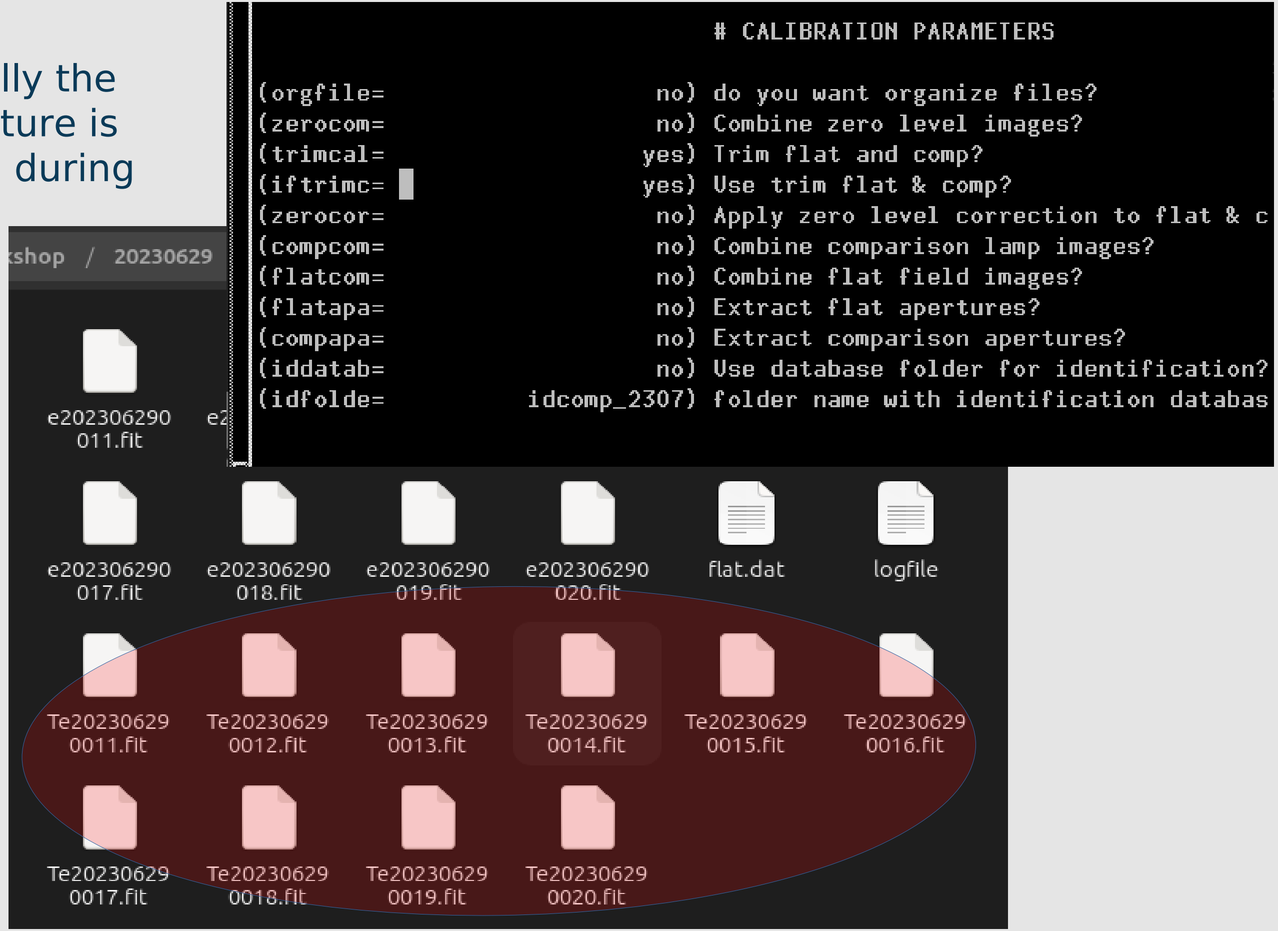
```
###### ZEROCOMBINE
if (access("zero/Zero.fit")){
    zerocomb=no
}
if (zerocomb==yes){
    cd "zero/"
    unlearn zerocombine
    zerocombine.reject="minmax"
    zerocombine.rdnoise= "READNOIS"
    zerocombine.gain    = "GAIN"
    zerocombine (input="@zero.dat",output="Zero.fit")
    cd "../"
}
```



“Zero.fit” selected (16.8 MB)

4) trim flat and comp

- completely optional, but usually the first and last pixel of each aperture is saturated and can be annoying during normalization.
- related with lines database 'idcomp'



```
                                  # CALIBRATION PARAMETERS

(orgfile=                     no) do you want organize files?
(zerocom=                     no) Combine zero level images?
(trimcal=                    yes) Trim flat and comp?
(iftrimc=                    yes) Use trim flat & comp?
(zerocor=                     no) Apply zero level correction to flat & c
(compcom=                     no) Combine comparison lamp images?
(flatcom=                     no) Combine flat field images?
(flatapa=                     no) Extract flat apertures?
(compapa=                     no) Extract comparison apertures?
(iddatab=                     no) Use database folder for identification?
(idfolde=            idcomp_2307) folder name with identification databas
```

zerocom = no

trimcal = yes

iftrimc = yes

:go

- new files, prefix **T**

Te*.fit

5) Bias correction → flat and comp

trimcal = no

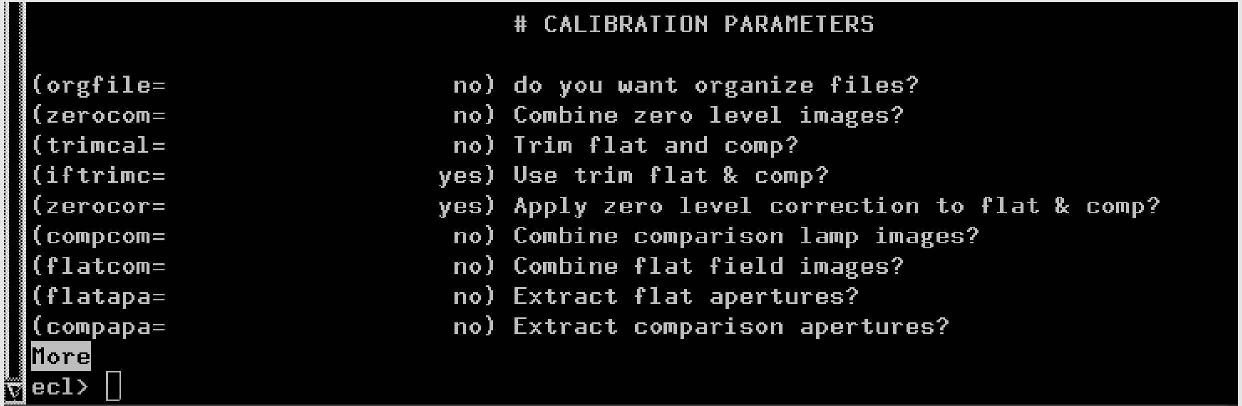
iftrimc = yes

zerocor = yes

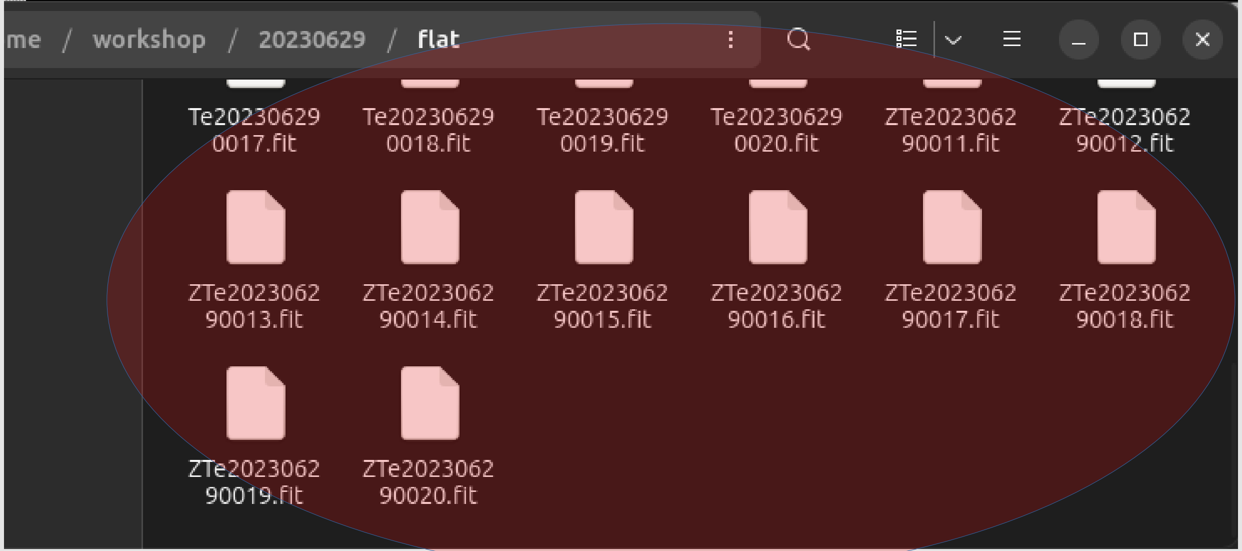
:go

- new files, prefix **Z**

ZTe*.fit



```
                                    # CALIBRATION PARAMETERS

(orgfile=                          no) do you want organize files?
(zerocom=                          no) Combine zero level images?
(trimcal=                          no) Trim flat and comp?
(iftrimc=                         yes) Use trim flat & comp?
(zerocor=                         yes) Apply zero level correction to flat & comp?
(compcom=                          no) Combine comparison lamp images?
(flatcom=                          no) Combine flat field images?
(flatapa=                          no) Extract flat apertures?
(compapa=                          no) Extract comparison apertures?
More
ecl>
```



```
Te20230629      Te20230629      Te20230629      Te20230629      ZTe2023062      ZTe2023062
0017.fit        0018.fit        0019.fit        0020.fit        90011.fit       90012.fit

ZTe2023062      ZTe2023062      ZTe2023062      ZTe2023062      ZTe2023062      ZTe2023062
90013.fit       90014.fit       90015.fit       90016.fit       90017.fit       90018.fit

ZTe2023062      ZTe2023062
90019.fit       90020.fit
```

6) Combine comp (ThAr)

iftrimc = yes (keep “yes”)

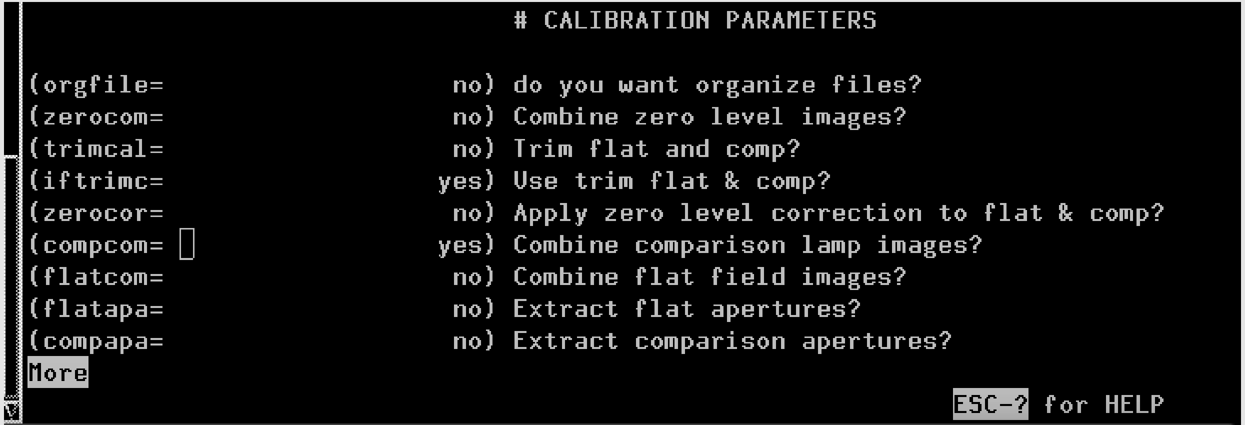
zerocor = no

compcom = yes

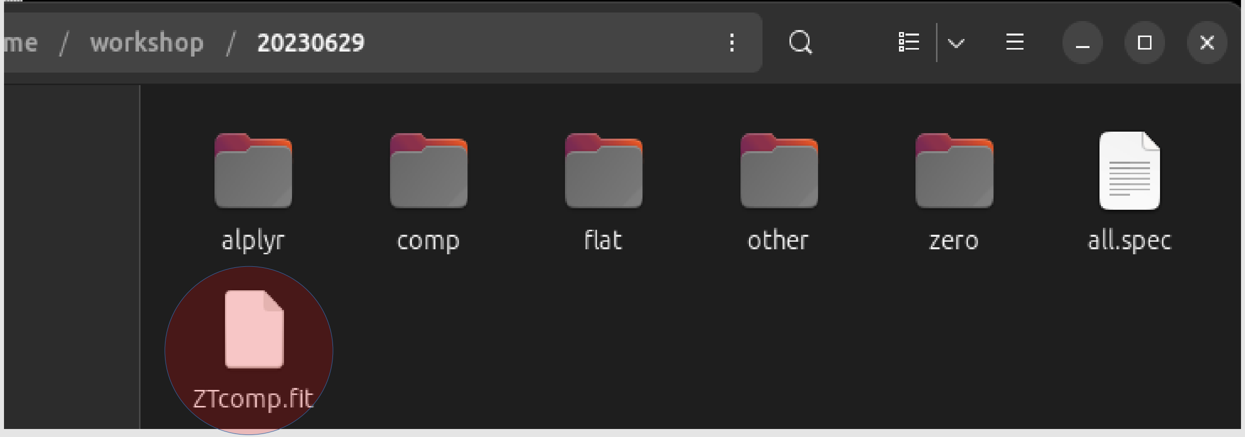
:go

- new file

ZTcomp.fit



```
                              # CALIBRATION PARAMETERS

(orgfile=                      no) do you want organize files?
(zerocom=                      no) Combine zero level images?
(trimcal=                      no) Trim flat and comp?
(iftrimc=                     yes) Use trim flat & comp?
(zerocor=                      no) Apply zero level correction to flat & comp?
(compcom=                     yes) Combine comparison lamp images?
(flatcom=                      no) Combine flat field images?
(flatapa=                      no) Extract flat apertures?
(compapa=                      no) Extract comparison apertures?
More
                                                              ESC-? for HELP
```



7) Combine flat field

iftrimc = yes

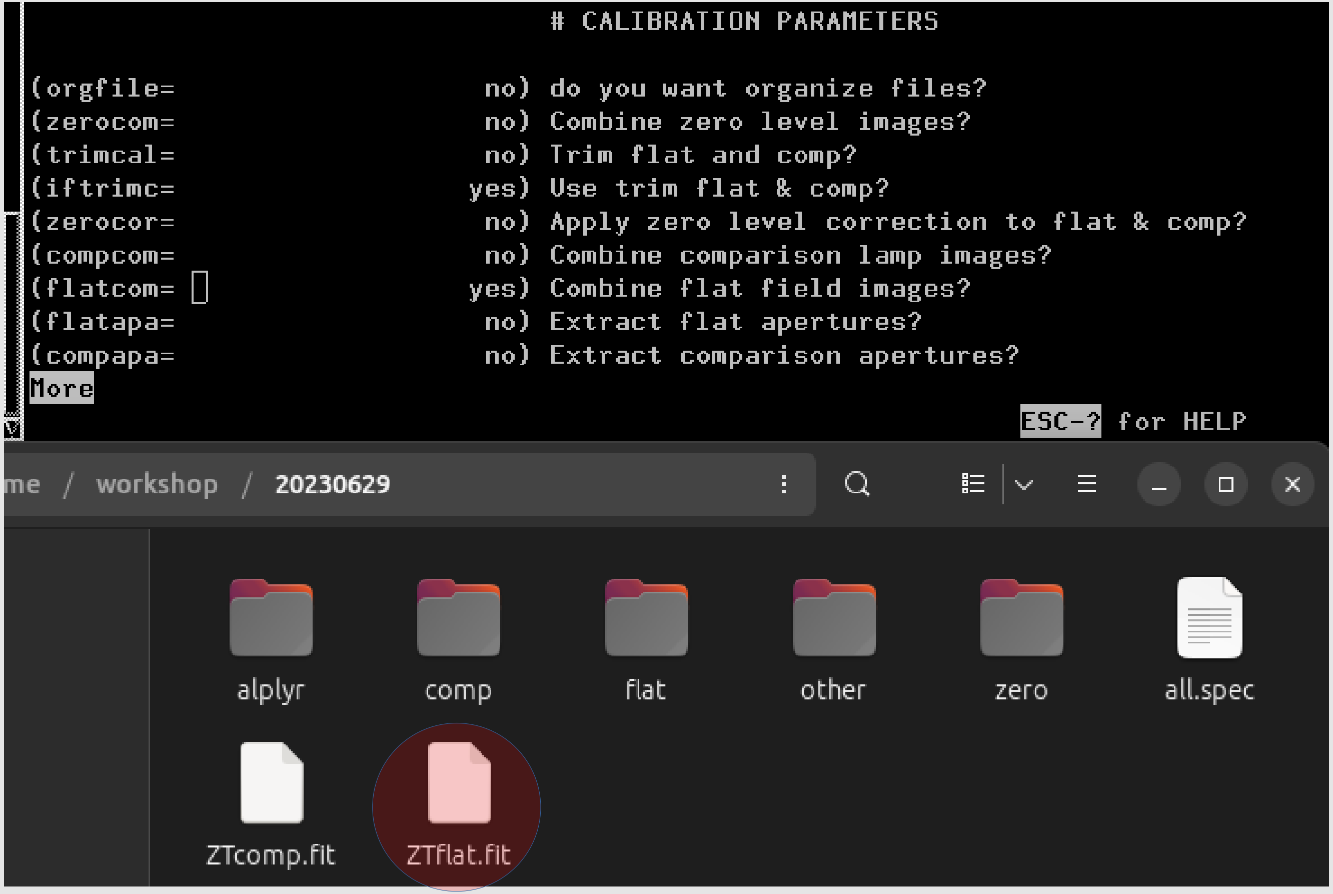
compcom = no

flatcom = yes

:go

- new file

ZTflat.fit



```
                               # CALIBRATION PARAMETERS

(orgfile=                        no) do you want organize files?
(zerocom=                        no) Combine zero level images?
(trimcal=                        no) Trim flat and comp?
(iftrimc=                       yes) Use trim flat & comp?
(zerocor=                        no) Apply zero level correction to flat & comp?
(compcom=                        no) Combine comparison lamp images?
(flatcom=                       yes) Combine flat field images?
(flatapa=                        no) Extract flat apertures?
(compapa=                        no) Extract comparison apertures?
More
                                                               ESC-? for HELP
```

8) Model apertures - Flat

iftrimc = yes

flatcom = no

flatapa = yes

:go

Find apertures for Ztflat? (yes):

Number of apertures to be found automatically (49):

Resize apertures for ZTflat? (yes):

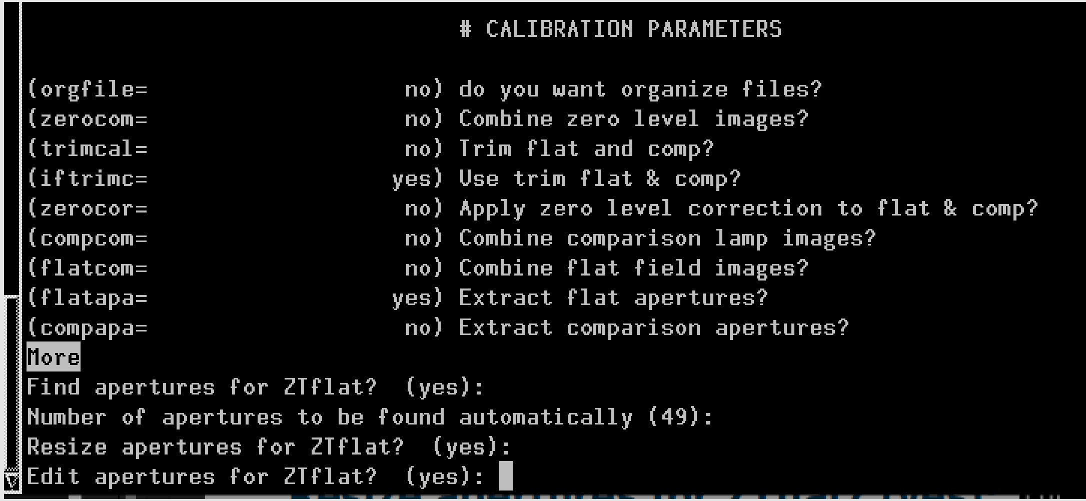
Edit apertures for Ztflat? (yes):

- accept everything with “enter” or typing “yes”. Default answer in parenthesis.

- value in parenthesis (yes/no) is Predefined. Press ENTER

- in order to do everything a bit easy/fast. We will use a database for wavelength calibration, for that we need to choose 49 apertures.

Background fitting is not necessarily because overlapping.



```
                                # CALIBRATION PARAMETERS

(orgfile=                      no) do you want organize files?
(zerocom=                      no) Combine zero level images?
(trimcal=                      no) Trim flat and comp?
(iftrimc=                     yes) Use trim flat & comp?
(zerocor=                      no) Apply zero level correction to flat & comp?
(compcom=                      no) Combine comparison lamp images?
(flatcom=                      no) Combine flat field images?
(flatapa=                     yes) Extract flat apertures?
(compapa=                      no) Extract comparison apertures?
More
Find apertures for ZTflat?  (yes):
Number of apertures to be found automatically (49):
Resize apertures for ZTflat?  (yes):
Edit apertures for ZTflat?  (yes):
```

CURSOR KEY - interactive

Good practice:

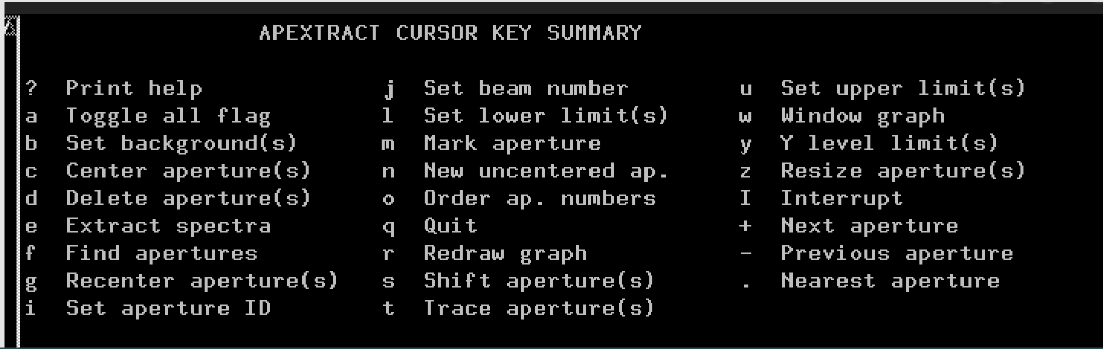
- **NEVER** resize the window directly when interactive is activated!
  if yes→ ':' resize → enter
- **NEVER** do click on the interactive window!
  if yes→ click on green square→ Delete

Help window: '?'
q+q → exit help

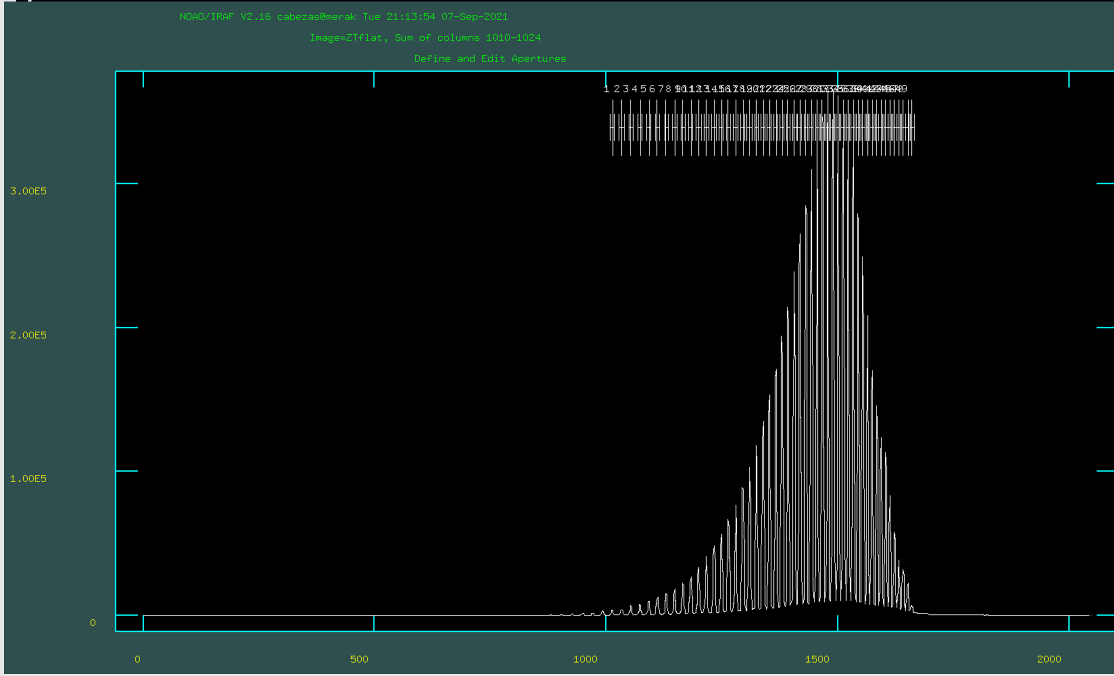
Window: w+e+e → Zoom bottom left/top right corner

w+a → redraw/ show all

w+u/d/l/r → move up/down/left/right



```
                        APEXTRACT CURSOR KEY SUMMARY

?  Print help               j  Set beam number         u  Set upper limit(s)
a  Toggle all flag          l  Set lower limit(s)      w  Window graph
b  Set background(s)        m  Mark aperture           y  Y level limit(s)
c  Center aperture(s)       n  New uncentered ap.      z  Resize aperture(s)
d  Delete aperture(s)       o  Order ap. numbers       I  Interrupt
e  Extract spectra          q  Quit                    +  Next aperture
f  Find apertures           r  Redraw graph            -  Previous aperture
g  Recenter aperture(s)     s  Shift aperture(s)       .  Nearest aperture
i  Set aperture ID          t  Trace aperture(s)
```



NOAO/IRAF V2.16 cabezas@merak Tue 21:13:54 07-Sep-2021

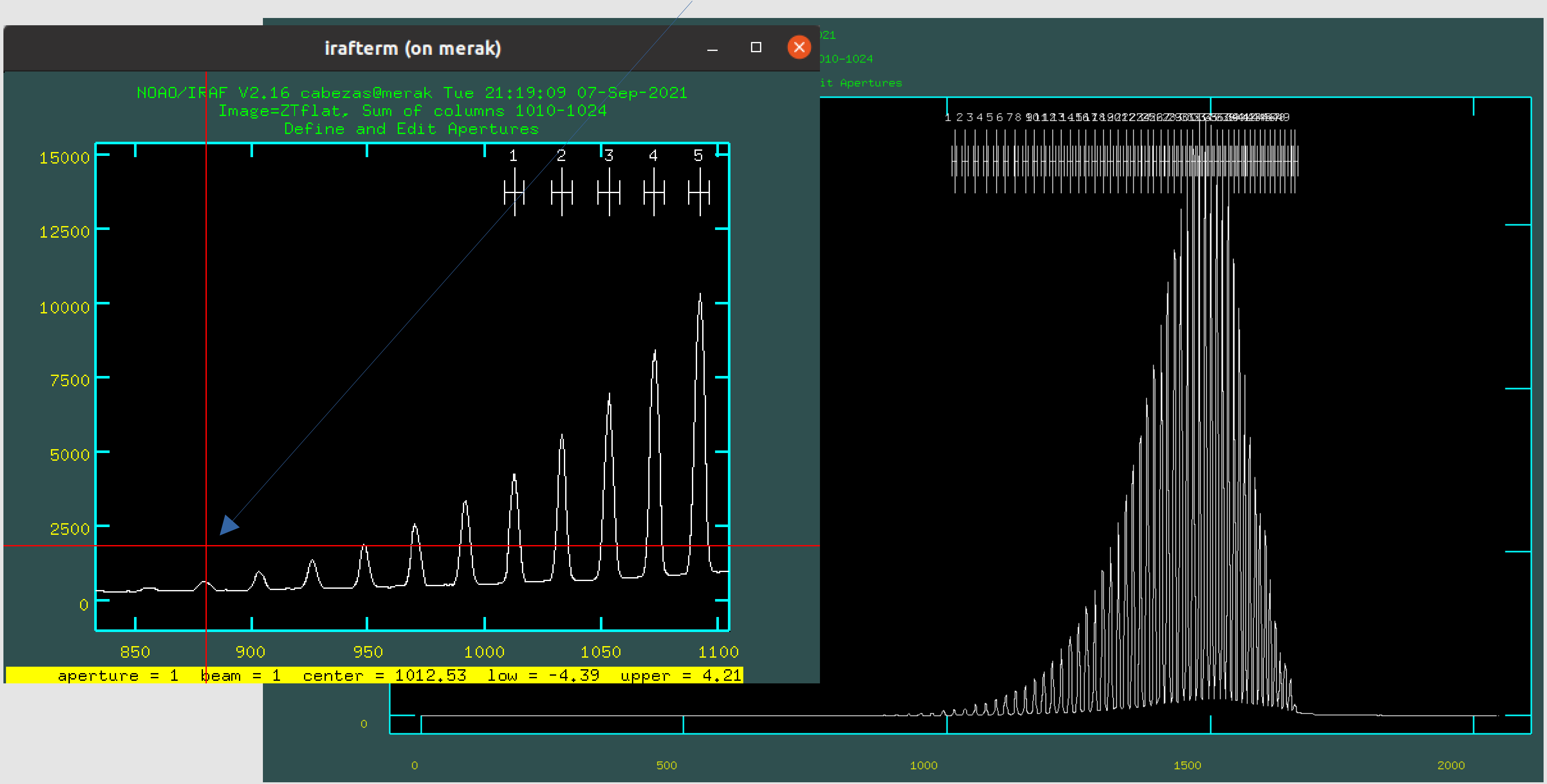
Image=ZTflat, Sum of columns 1010-1024

Define and Edit Apertures

8) Model apertures - Flat

mark: **m**

- first aperture near the pixel ~860-890
→ 6 additional apertures



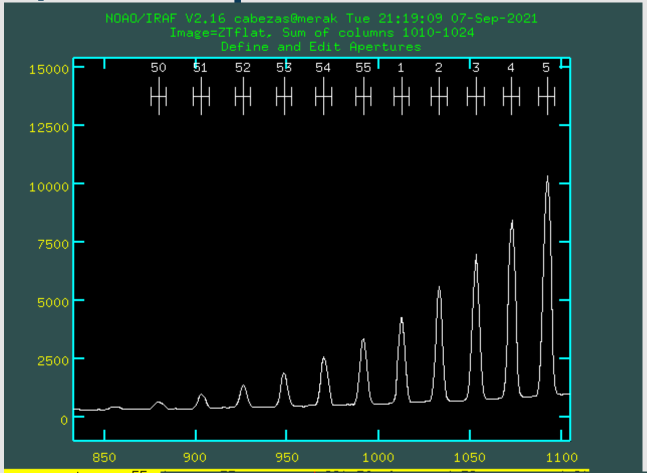
irafterm (on merak)

```
NOAO/IRAF V2.16 cabezas@merak Tue 21:19:09 07-Sep-2021
Image=ZTflat, Sum of columns 1010-1024
Define and Edit Apertures
```

```
aperture = 1   beam = 1   center = 1012.53   low = -4.39   upper = 4.21
```

8) Model apertures - Flat

w+a → redraw/ show all
w+e+e→ zoom



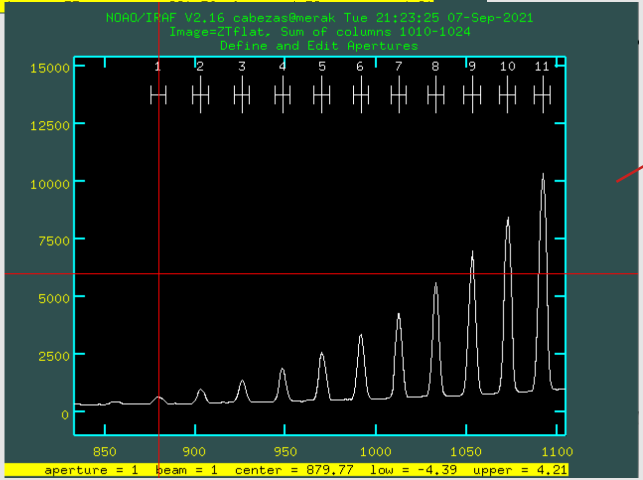
```
NOAO/IRAF V2.16 cabezas@merak Tue 21:19:09 07-Sep-2021
Image=ZTflat, Sum of columns 1010-1024
Define and Edit Apertures
```

```
aperture = 55
```

First aperture
Number 1,
no 50

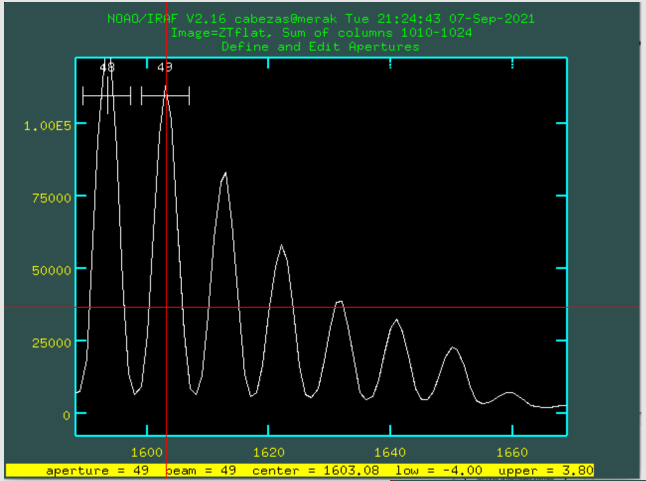
Sort: **o**

'1'
enter



```
NOAO/IRAF V2.16 cabezas@merak Tue 21:23:25 07-Sep-2021
Image=ZTflat, Sum of columns 1010-1024
Define and Edit Apertures
```

```
aperture = 1  beam = 1  center = 879.77  low = -4.39  upper = 4.21
```



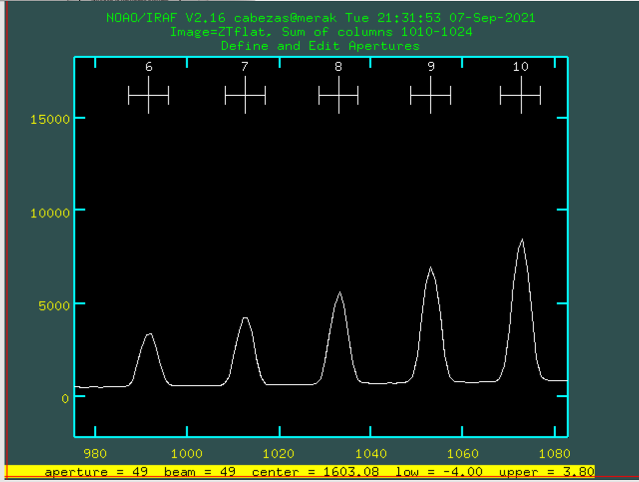
```
NOAO/IRAF V2.16 cabezas@merak Tue 21:24:43 07-Sep-2021
Image=ZTflat, Sum of columns 1010-1024
Define and Edit Apertures
```

```
aperture = 49  beam = 49  center = 1603.08  low = -4.00  upper = 3.80
```

- delete extra apertures
We need only 49
→ over the aperture
Then : **d**

- check apertures,
Zoom in/out and
Aperture numbers

All right? → **'q'**
'Trace apertures for Ztflat?
→ (yes)



```
NOAO/IRAF V2.16 cabezas@merak Tue 21:31:53 07-Sep-2021
Image=ZTflat, Sum of columns 1010-1024
Define and Edit Apertures
```

```
aperture = 49  beam = 49  center = 1603.08  low = -4.00  upper = 3.80
```

8) Model apertures - Flat

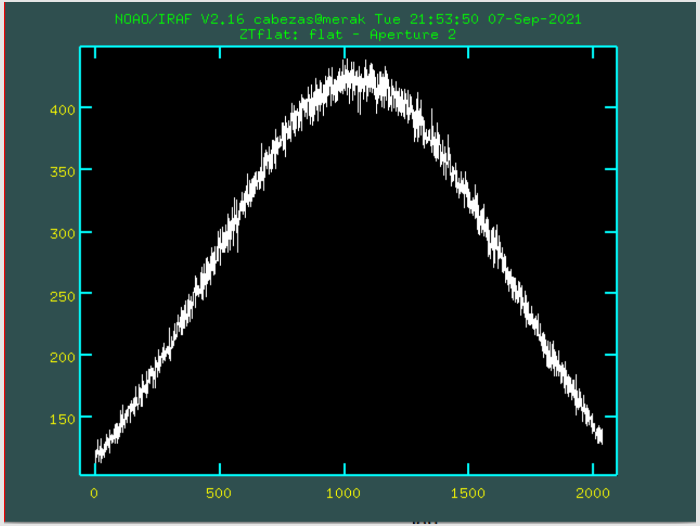
Fit curve to aperture N of Ztflat interactively (yes): YES

Write apertures for ZTflat to database (yes):

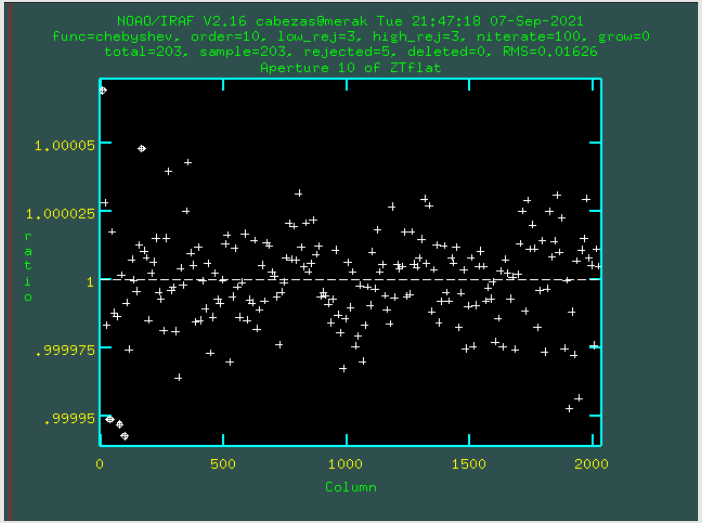
Extract aperture for ZTflat? (yes):

Review extracted spectra from ZTflat? (yes): YES

Review extracted spectrum for aperture 1 from ZTflat? (yes)
YES



```
NOAO/IRAF V2.16 cabezas@merak Tue 21:53:50 07-Sep-2021
ZTflat: flat - Aperture 2
```

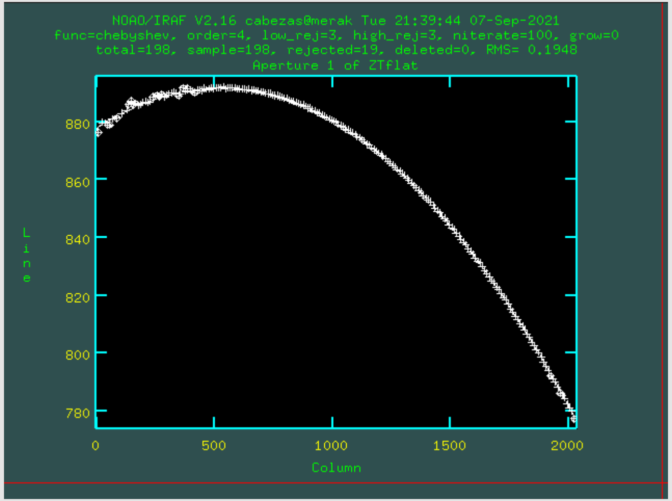


```
NOAO/IRAF V2.16 cabezas@merak Tue 21:47:18 07-Sep-2021
func=chebyshev, order=10, low_rej=3, high_rej=3, niterate=100, grow=0
total=203, sample=203, rejected=5, deleted=0, RMS=0.01626
Aperture 10 of ZTflat
```

Polinomial fitting of the echelle orders

- change order **:o N**
- change function: **:f spline3/legendre/ chebyshev**

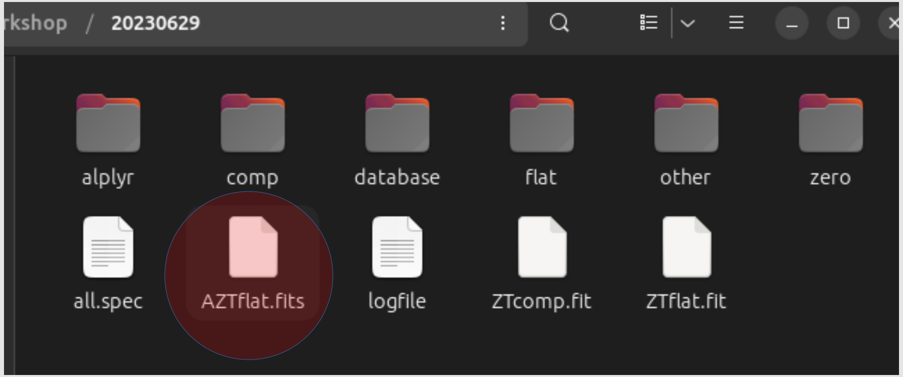
- relative residuals **k**
- residuals (px) **j**
- aperture **h**
- fit **f**

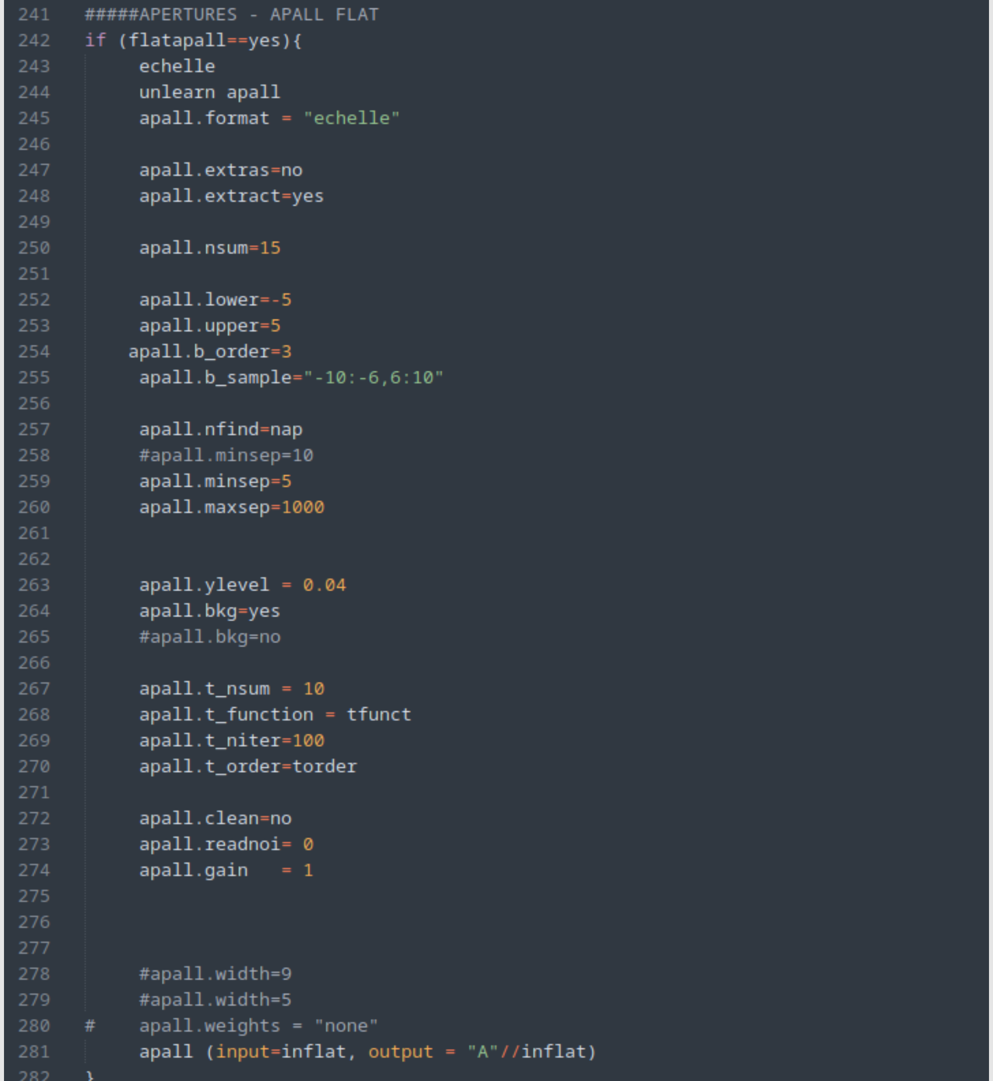


```
NOAO/IRAF V2.16 cabezas@merak Tue 21:39:44 07-Sep-2021
func=chebyshev, order=4, low_rej=3, high_rej=3, niterate=100, grow=0
total=198, sample=198, rejected=19, deleted=0, RMS= 0.1948
Aperture 1 of ZTflat
```

8) Model apertures - Flat

- new file: **AZTflat.fit**





```
#####APERTURES - APALL FLAT
if (flatapall==yes){
    echelle
    unlearn apall
    apall.format = "echelle"

    apall.extras=no
    apall.extract=yes

    apall.nsum=15

    apall.lower=-5
    apall.upper=5
   apall.b_order=3
    apall.b_sample="-10:-6,6:10"

    apall.nfind=nap
    #apall.minsep=10
    apall.minsep=5
    apall.maxsep=1000


    apall.ylevel = 0.04
    apall.bkg=yes
    #apall.bkg=no

    apall.t_nsum = 10
    apall.t_function = tfunct
    apall.t_niter=100
    apall.t_order=torder

    apall.clean=no
    apall.readnoi= 0
    apall.gain   = 1



    #apall.width=9
    #apall.width=5
#   apall.weights = "none"
    apall (input=inflat, output = "A"//inflat)
}
```

9) Model apertures – Comparison lamp

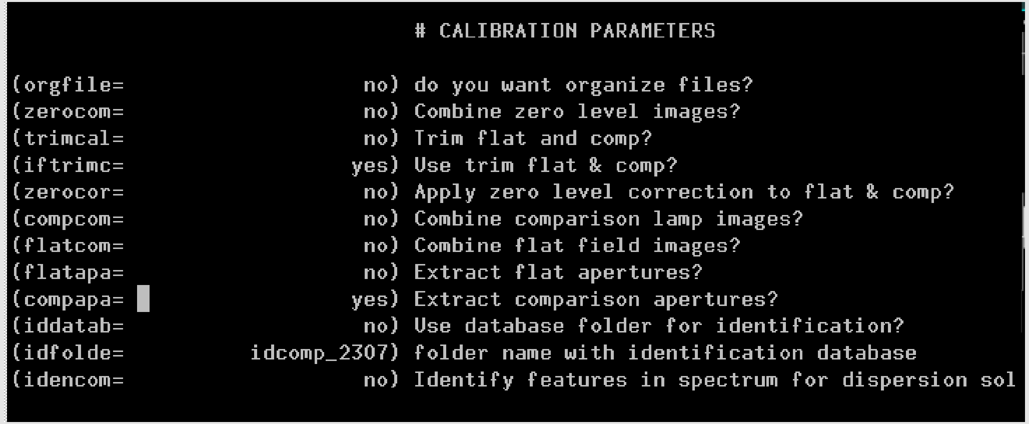
- Template: AZTflat.fit

iftrimc = yes

flatapa = no

compapa = yes

:go



```
                          # CALIBRATION PARAMETERS

(orgfile=                    no) do you want organize files?
(zerocom=                    no) Combine zero level images?
(trimcal=                    no) Trim flat and comp?
(iftrimc=                   yes) Use trim flat & comp?
(zerocor=                    no) Apply zero level correction to flat & comp?
(compcom=                    no) Combine comparison lamp images?
(flatcom=                    no) Combine flat field images?
(flatapa=                    no) Extract flat apertures?
(compapa=                   yes) Extract comparison apertures?
(iddatab=                    no) Use database folder for identification?
(idfolde=           idcomp_2307) folder name with identification database
(idencom=                    no) Identify features in spectrum for dispersion sol
```

Edit apertures for ZTcomp? (yes):

Usually “no”, but better to check once!

q

Write apertures for ZTcomp to database (yes):

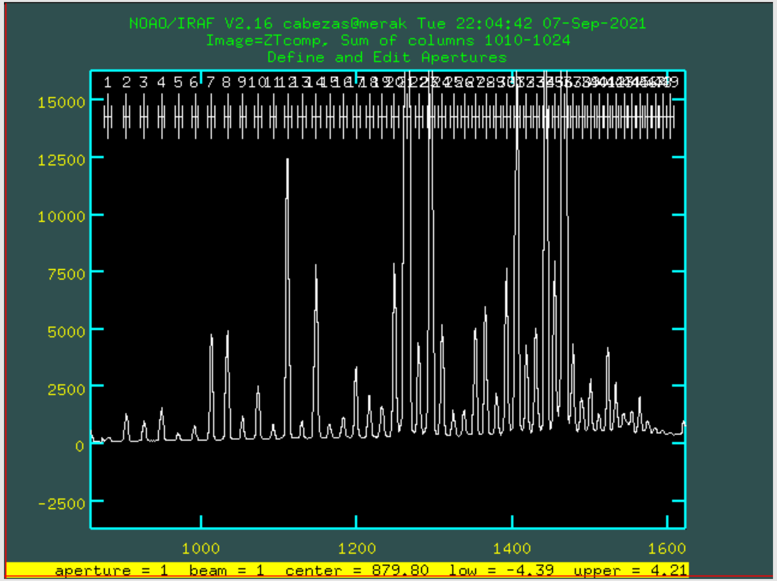
Extract aperture spectra for ZTcomp? (yes):

Review extracted spectra from ZTcomp? (yes):

Review extracted spectrum for aperture 1 from ZTcomp? (yes) YES

- if you are using database, the first aperture has to correspond to the first aperture in the atlas.

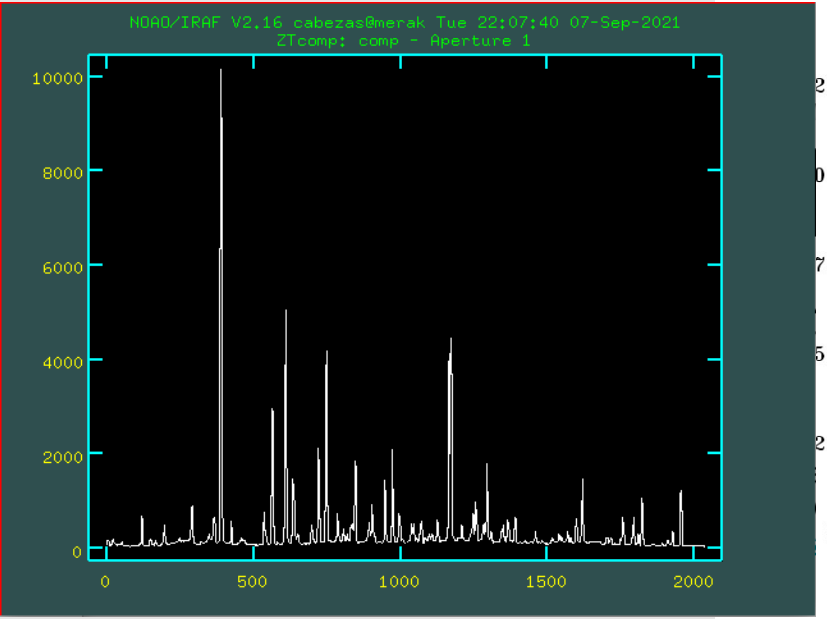
Inside folder **idcomp**



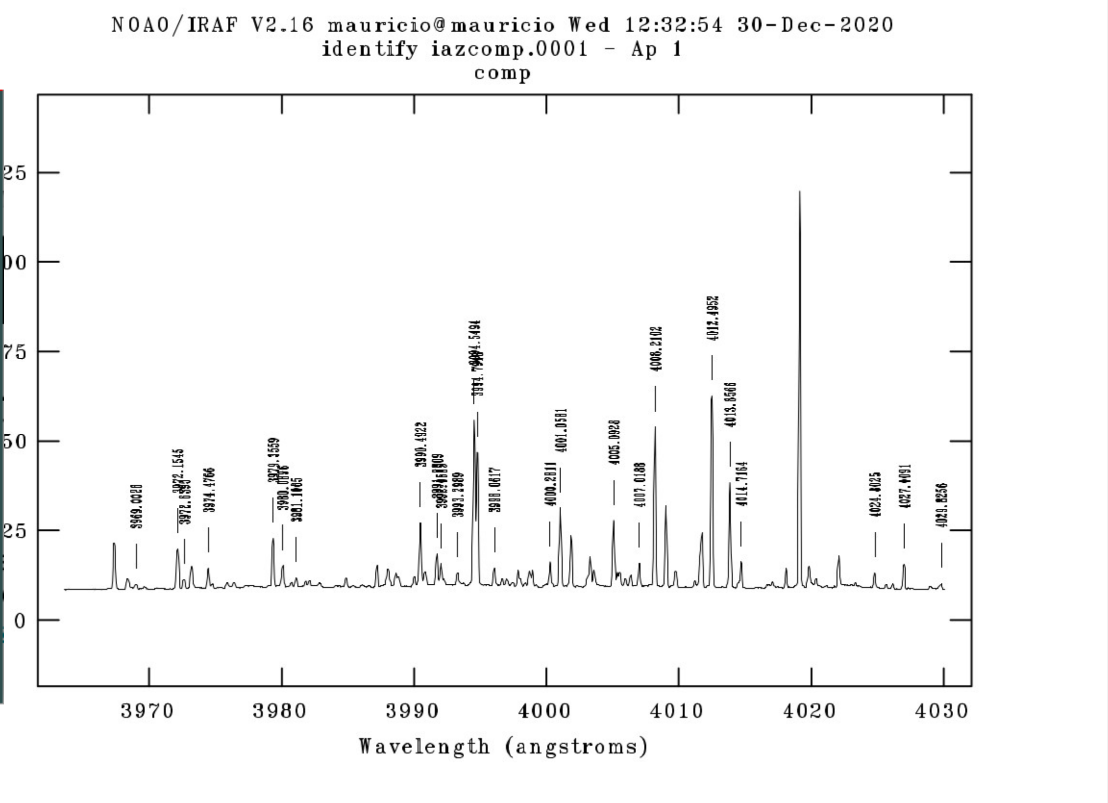
9) Model apertures – Comparison lamp

Usually not necessary: look at atlas
in idcomp folder

```
okular oesatlas.pdf
```



NOAO/IRAF V2.16 cabezas@merak Tue 22:07:40 07-Sep-2021
ZTcomp: comp - Aperture 1

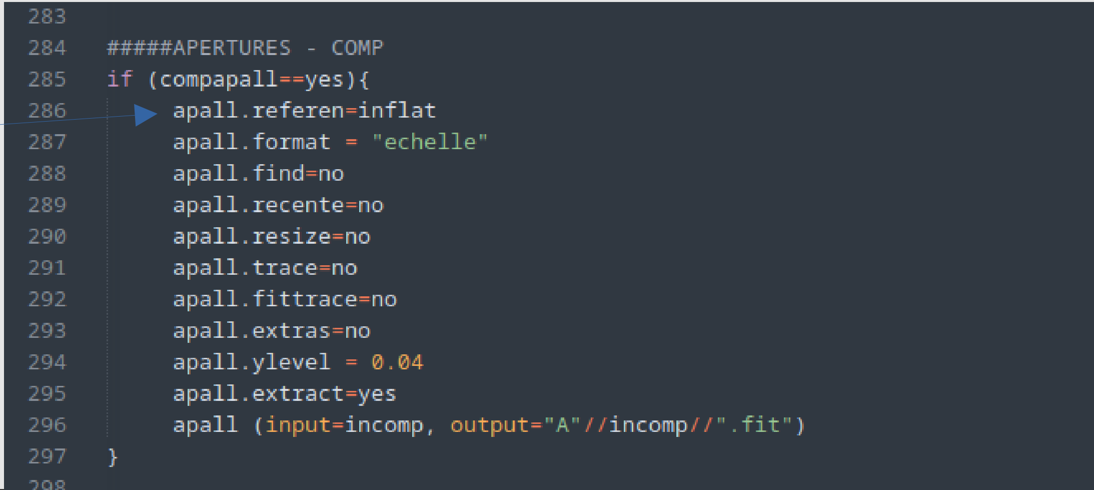


NOAO/IRAF V2.16 mauricio@mauricio Wed 12:32:54 30-Dec-2020
identify iazcomp.0001 - Ap 1
comp

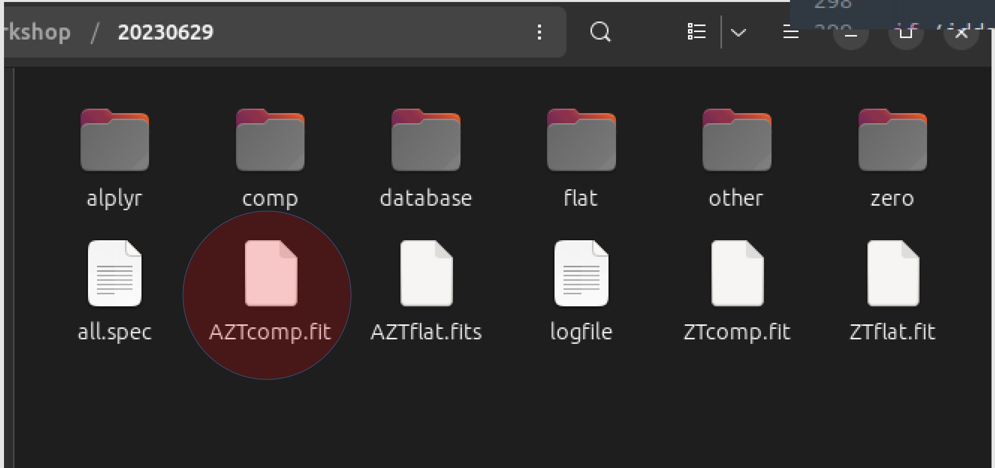
9) Model apertures – Comparison lamp

- reference → **AZTflat**

- new file: **AZTcomp.fit**



```
#####APERTURES - COMP
if (compapall==yes){
        apall.referen=inflat
        apall.format = "echelle"
        apall.find=no
        apall.recente=no
        apall.resize=no
        apall.trace=no
        apall.fittrace=no
        apall.extras=no
        apall.ylevel = 0.04
        apall.extract=yes
        apall (input=incomp, output="A"//incomp//".fit")
}
```



10) Wavelength calibration

- using database, folder “idcomp” must be in the main path

  iftrimc = yes

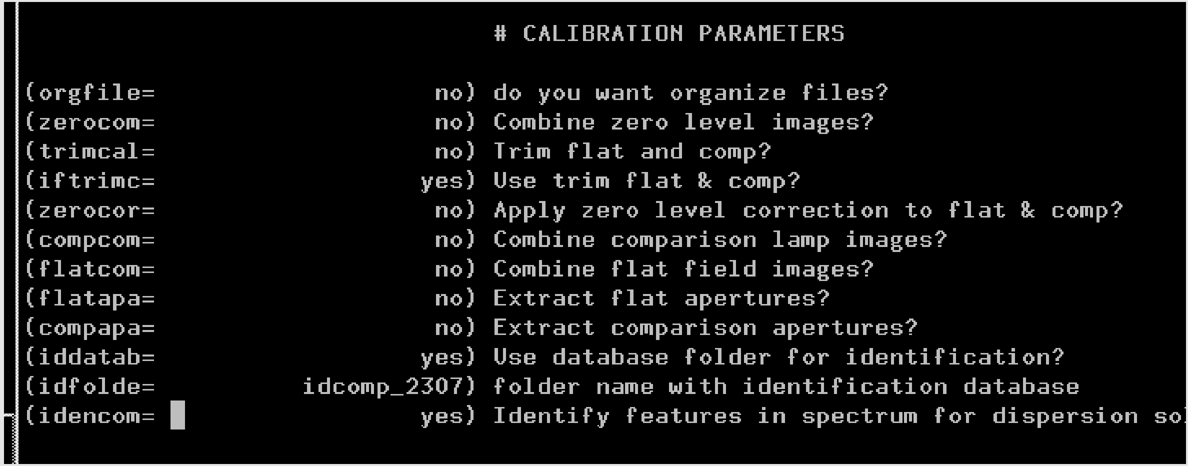
  compapa = no

  iddatab = yes

  idfolder = idcomp_2307

  idencom = yes

  :go



```
                                # CALIBRATION PARAMETERS

(orgfile=                   no) do you want organize files?
(zerocom=                   no) Combine zero level images?
(trimcal=                   no) Trim flat and comp?
(iftrimc=                  yes) Use trim flat & comp?
(zerocor=                   no) Apply zero level correction to flat & comp?
(compcom=                   no) Combine comparison lamp images?
(flatcom=                   no) Combine flat field images?
(flatapa=                   no) Extract flat apertures?
(compapa=                   no) Extract comparison apertures?
(iddatab=                  yes) Use database folder for identification?
(idfolde=          idcomp_2307) folder name with identification database
(idencom=                  yes) Identify features in spectrum for dispersion so
```

- in database, lines can be shifted few pixels, in order to fix it we need to “shift” the lines of our database. If doesn’t work, recommend delete every line with **d**, and mark new lines **(m)** comparing with the atlas.

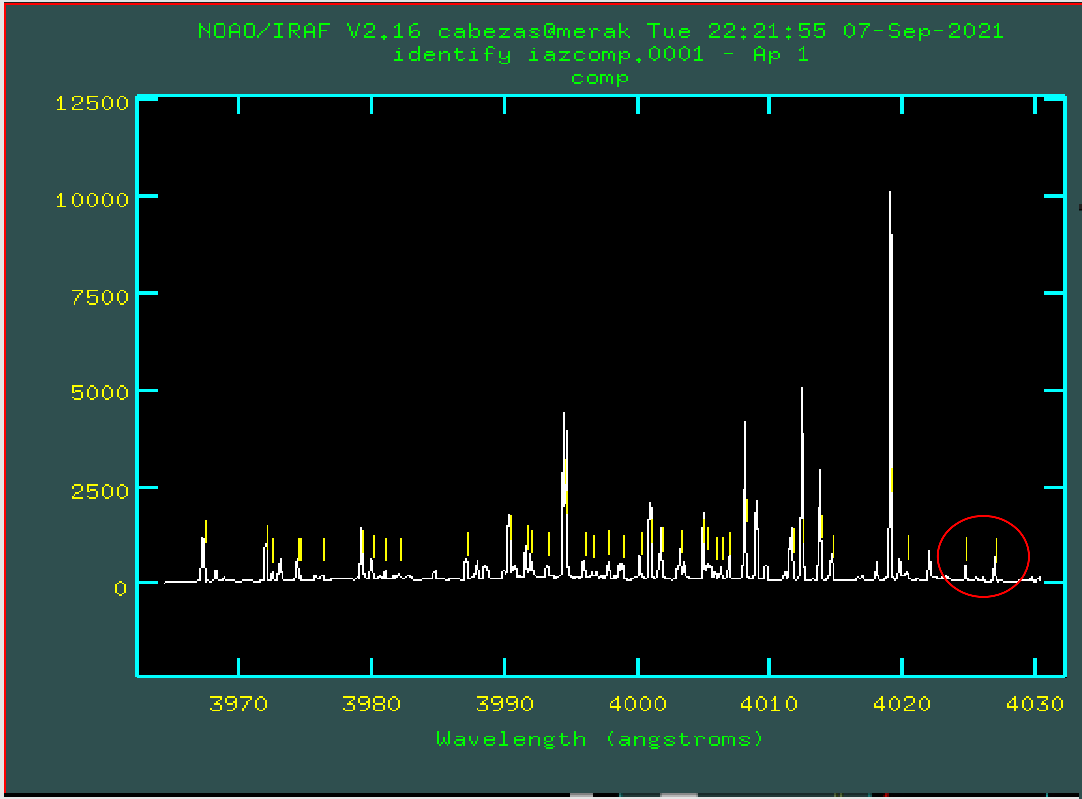
  Shift → **s**

- always fit!

  f (delete point **d**), **q, q**

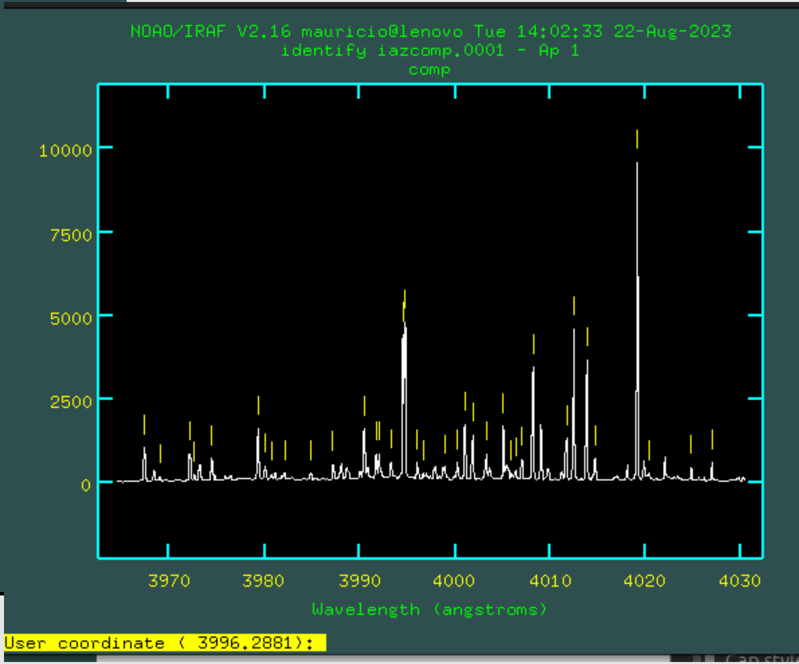
- rms ~ 0.007 acceptable

- **Write feature data to the database (yes)?**



```
NOAO/IRAF V2.16 cabezas@merak Tue 22:21:55 07-Sep-2021
identify iazcomp.0001 - Ap 1
comp
```

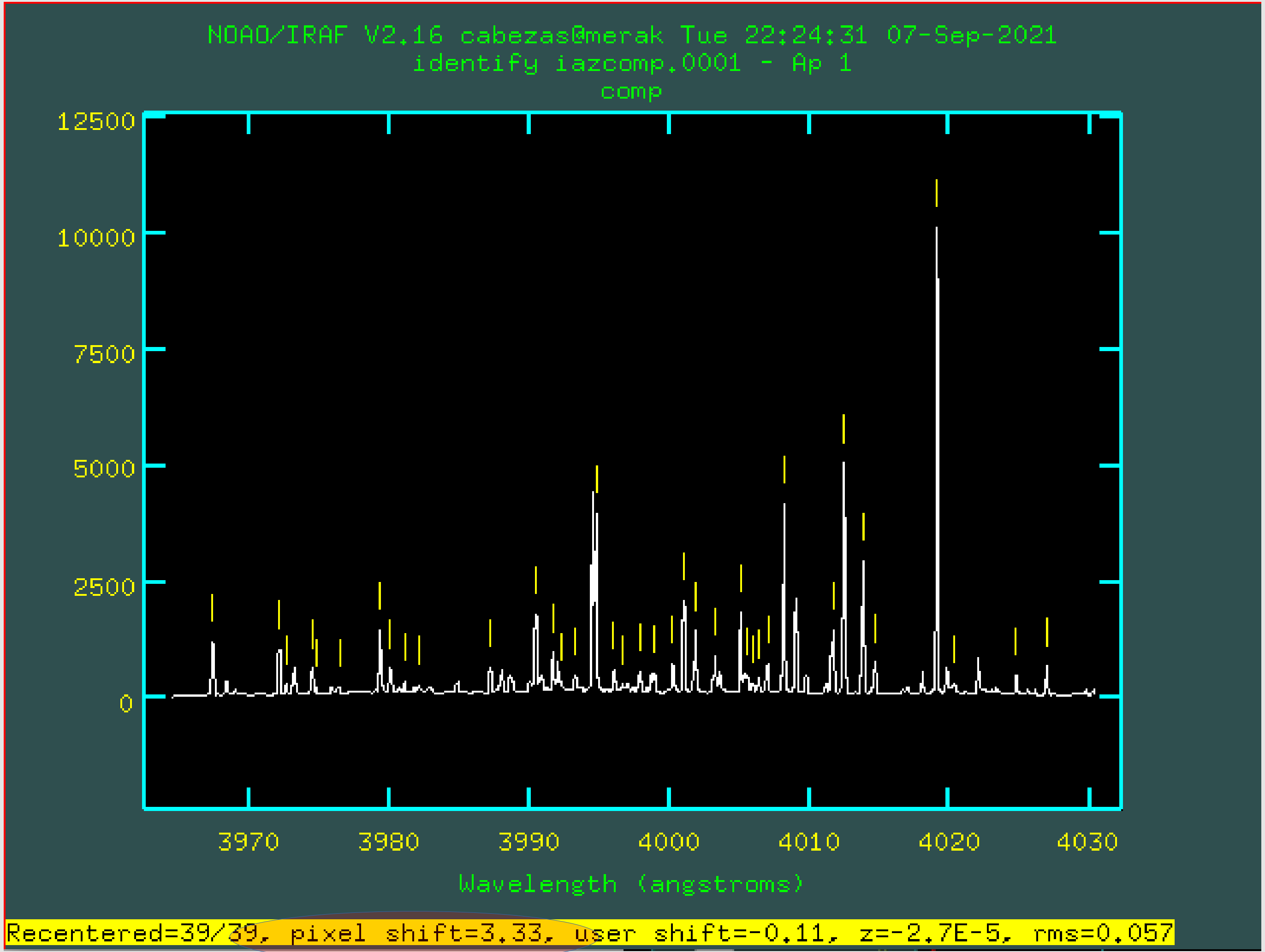
No match!
Press **"s"**



```
NOAO/IRAF V2.16 mauricio@lenovo Tue 14:02:33 22-Aug-2023
identify iazcomp.0001 - Ap 1
comp
```

```
User coordinate ( 3996.2881):
```

: enter

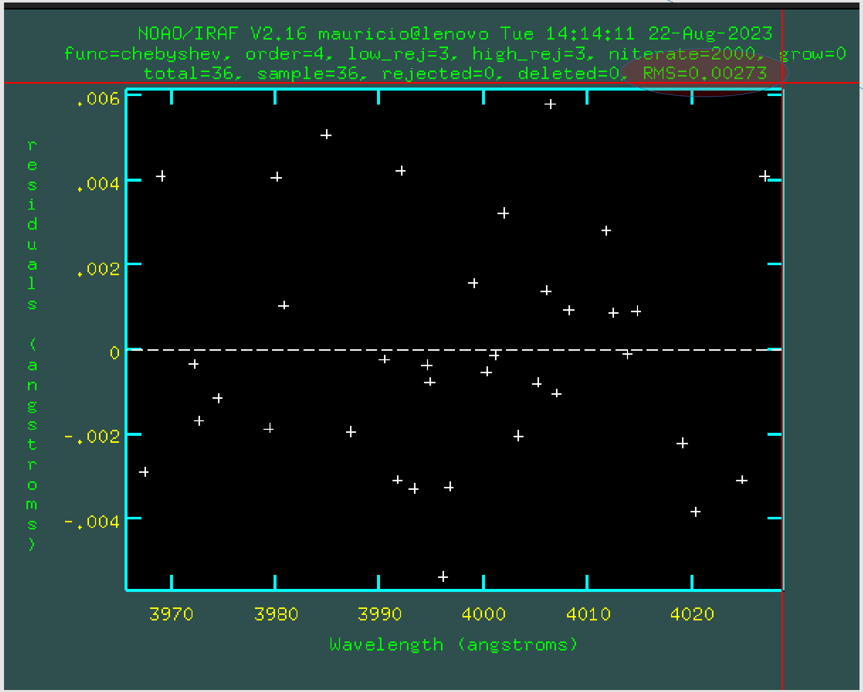


NOAO/IRAF V2.16 cabezas@merak Tue 22:24:31 07-Sep-2021
identify iazcomp.0001 - Ap 1
comp

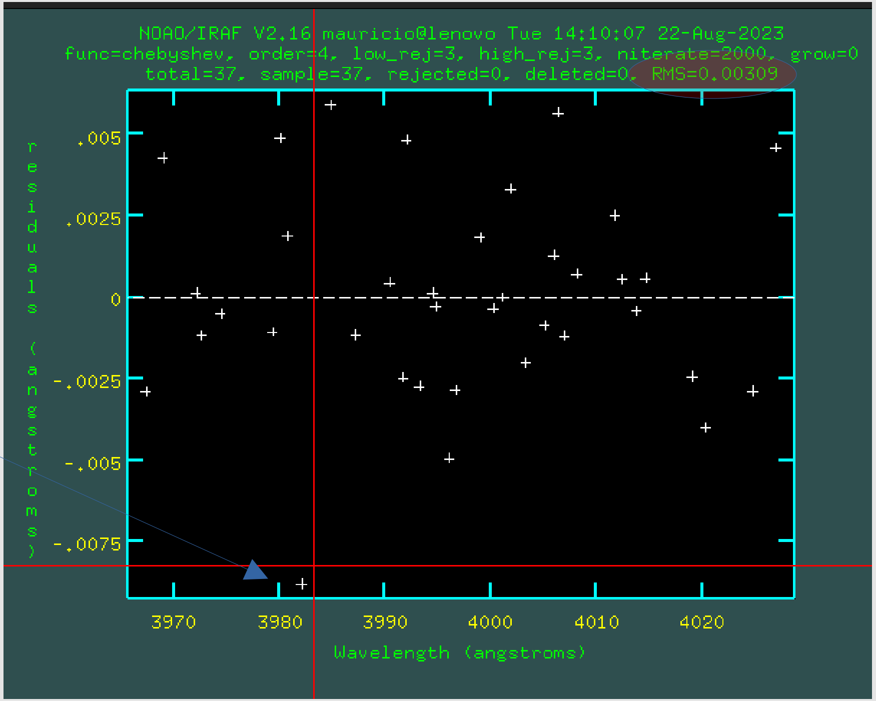
```
Recentered=39/39, pixel shift=3.33, user shift=-0.11, z=-2.7E-5, rms=0.057
```

10) Wavelength calibration - fit

- NO pattern in residuals
- no high order is recommended
- **d** to eliminate point
- **q, f** to refit and update plot



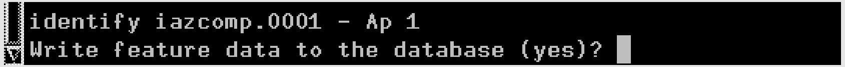
```
NOAO/IRAF V2.16 mauricio@lenovo Tue 14:14:11 22-Aug-2023
func=chebyshev, order=4, low_rej=3, high_rej=3, niterate=2000, grow=0
total=36, sample=36, rejected=0, deleted=0, RMS=0.00273
```



```
NOAO/IRAF V2.16 mauricio@lenovo Tue 14:10:07 22-Aug-2023
func=chebyshev, order=4, low_rej=3, high_rej=3, niterate=2000, grow=0
total=37, sample=37, rejected=0, deleted=0, RMS=0.00309
```

- better RMS, no significant

→ **q,q** to the next aperture



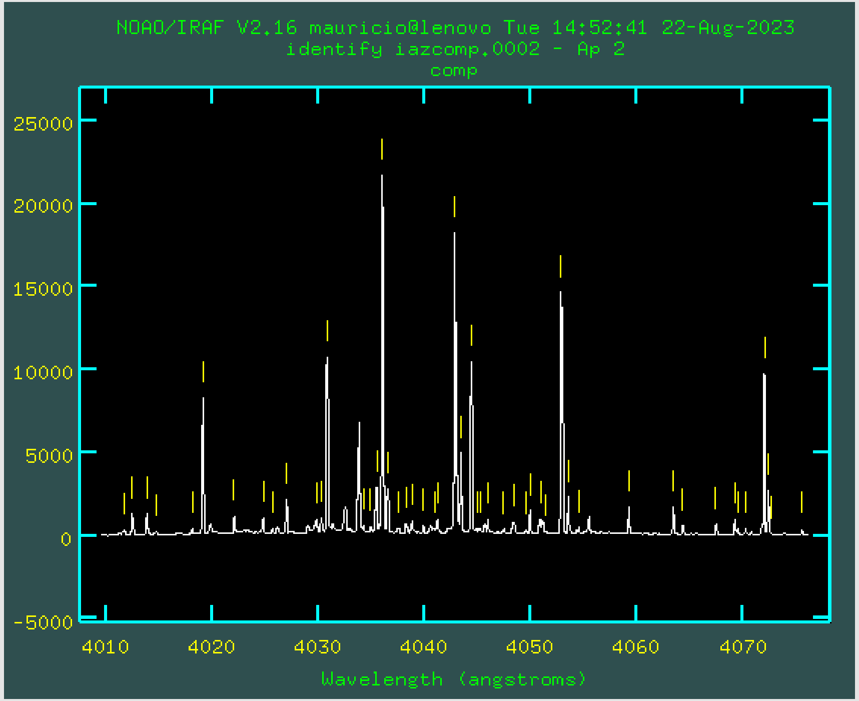
```
identify iazcomp.0001 - Ap 1
Write feature data to the database (yes)?
```

- yes, 49 times :)

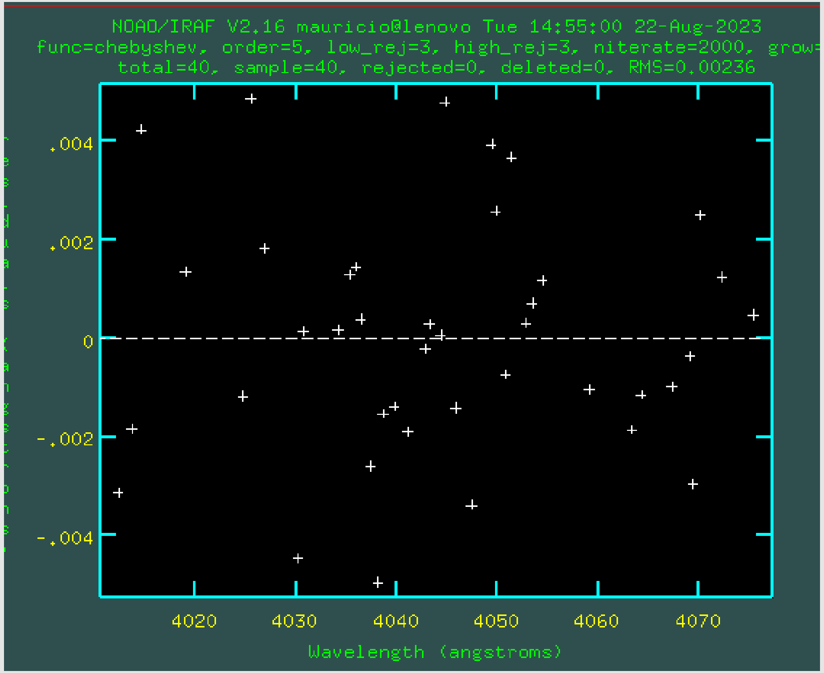
10) Wavelength calibration – bad match?? :(

- **L**, automatic mark
- **f,** refit
- ‘clean’

→ happy, **q,q**



```
NOAO/IRAF V2.16 mauricio@lenovo Tue 14:52:41 22-Aug-2023
identify iazcomp.0002 - Ap 2
comp
```

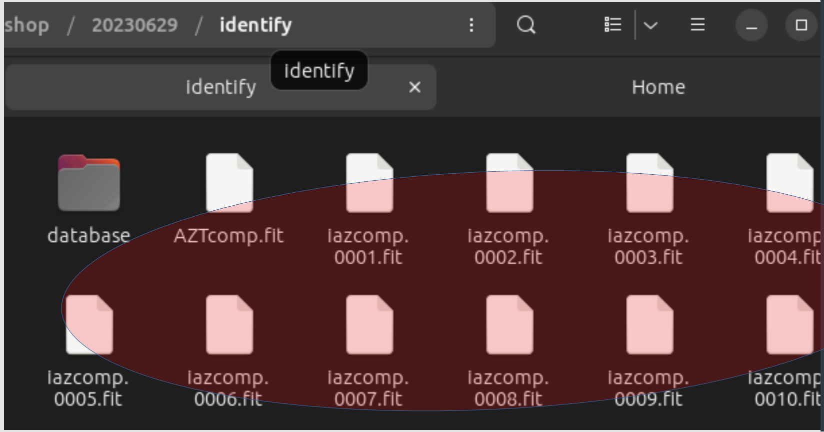


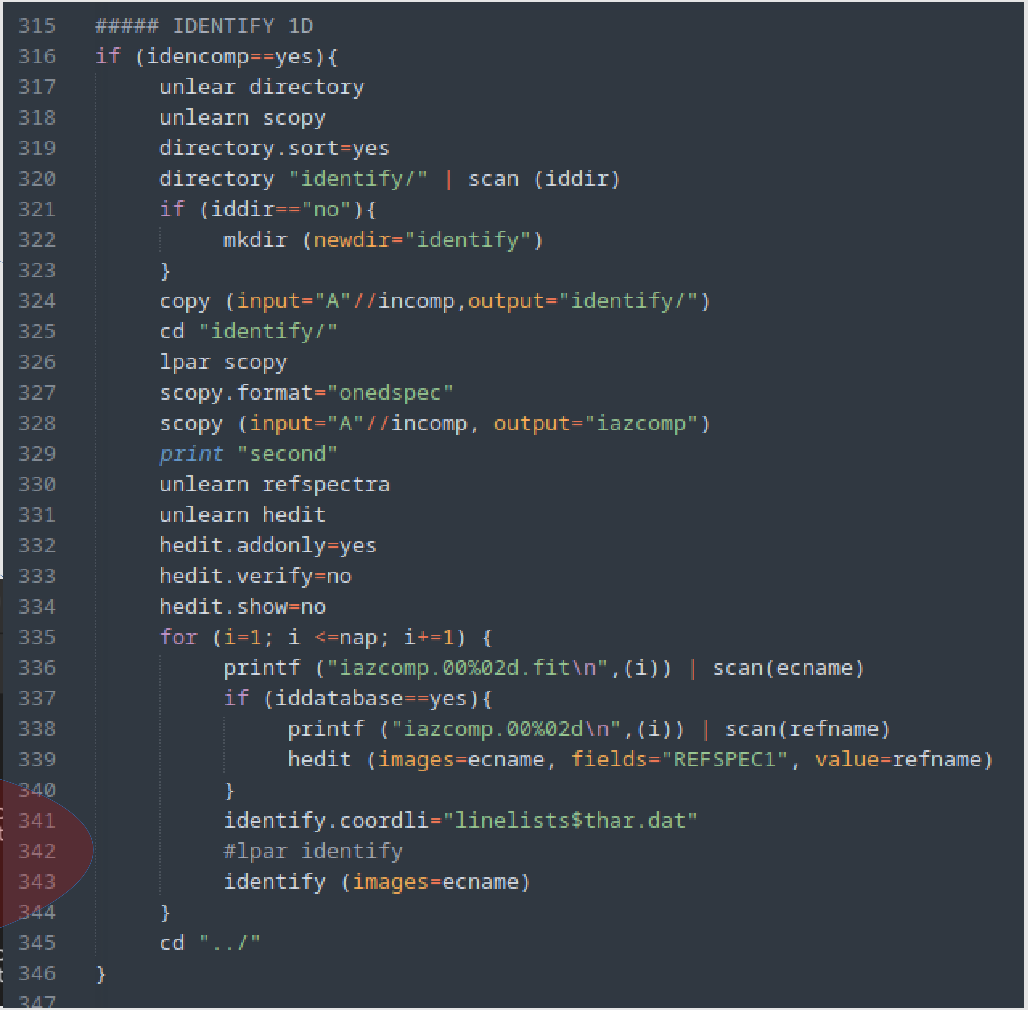
```
NOAO/IRAF V2.16 mauricio@lenovo Tue 14:55:00 22-Aug-2023
func=chebyshev, order=5, low_rej=3, high_rej=3, niterate=2000, grow=
total=40, sample=40, rejected=0, deleted=0, RMS=0.00236
```

10) Wavelength calibration

- using the database
- for each aperture a file is created with the calibration information, which later will use as reference to the object’s apertures.

- new files: **iazcomp.00**.fit**



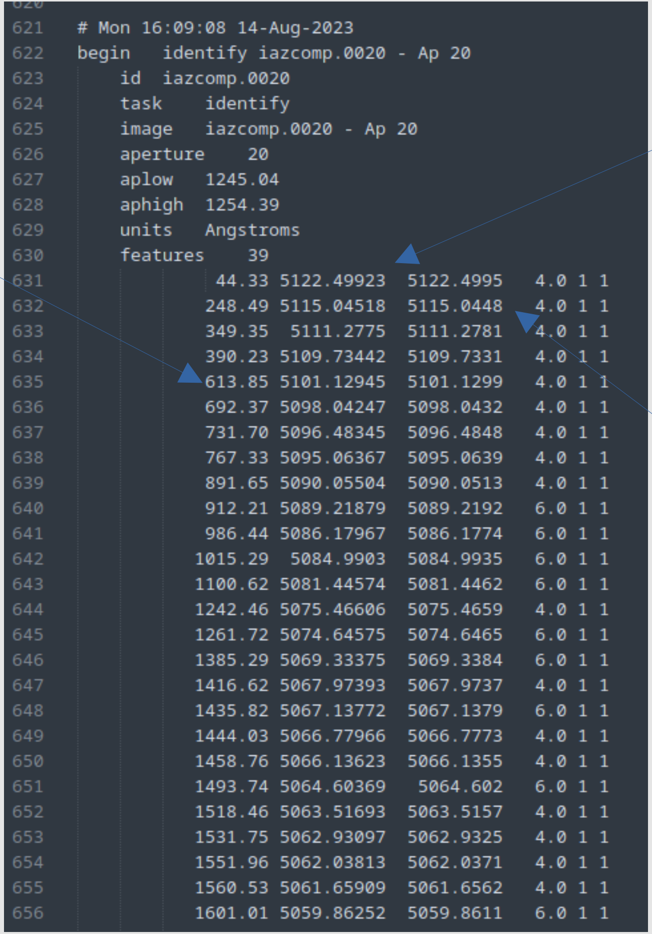


```
##### IDENTIFY 1D
if (idencomp==yes){
    unlear directory
    unlearn scopy
    directory.sort=yes
    directory "identify/" | scan (iddir)
    if (iddir=="no"){
        mkdir (newdir="identify")
    }
    copy (input="A"//incomp,output="identify/")
    cd "identify/"
    lpar scopy
    scopy.format="onedspec"
    scopy (input="A"//incomp, output="iazcomp")
    print "second"
    unlearn refspectra
    unlearn hedit
    hedit.addonly=yes
    hedit.verify=no
    hedit.show=no
    for (i=1; i <=nap; i+=1) {
        printf ("iazcomp.00%02d.fit\n",(i)) | scan(ecname)
        if (iddatabase==yes){
            printf ("iazcomp.00%02d\n",(i)) | scan(refname)
            hedit (images=ecname, fields="REFSPEC1", value=refname)
        }
        identify.coordli="linelists$thar.dat"
        #lpar identify
        identify (images=ecname)
    }
    cd "../"
}
```

10) Wavelength calibration – idcomp_* database

- pixel

- Last one is the valid



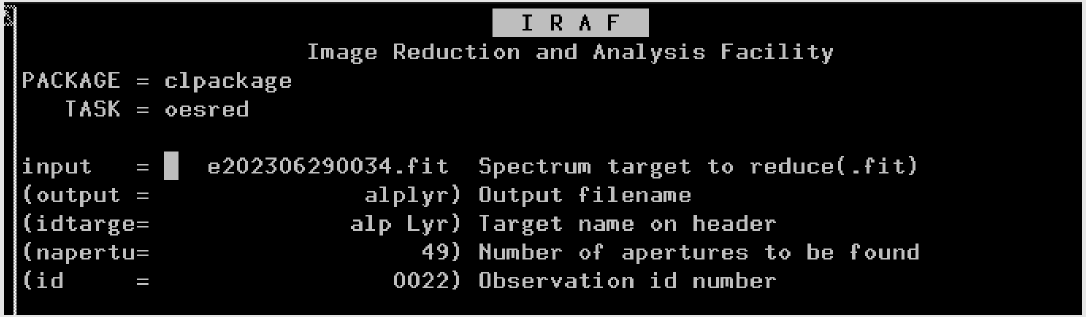
```
# Mon 16:09:08 14-Aug-2023
begin   identify iazcomp.0020 - Ap 20
    id  iazcomp.0020
    task    identify
    image   iazcomp.0020 - Ap 20
    aperture    20
    aplow   1245.04
    aphigh  1254.39
    units   Angstroms
    features    39
             44.33 5122.49923  5122.4995   4.0 1 1
            248.49 5115.04518  5115.0448   4.0 1 1
            349.35  5111.2775  5111.2781   4.0 1 1
            390.23 5109.73442  5109.7331   4.0 1 1
            613.85 5101.12945  5101.1299   4.0 1 1
            692.37 5098.04247  5098.0432   4.0 1 1
            731.70 5096.48345  5096.4848   4.0 1 1
            767.33 5095.06367  5095.0639   4.0 1 1
            891.65 5090.05504  5090.0513   4.0 1 1
            912.21 5089.21879  5089.2192   6.0 1 1
            986.44 5086.17967  5086.1774   6.0 1 1
           1015.29  5084.9903  5084.9935   6.0 1 1
           1100.62 5081.44574  5081.4462   6.0 1 1
           1242.46 5075.46606  5075.4659   4.0 1 1
           1261.72 5074.64575  5074.6465   6.0 1 1
           1385.29 5069.33375  5069.3384   6.0 1 1
           1416.62 5067.97393  5067.9737   4.0 1 1
           1435.82 5067.13772  5067.1379   6.0 1 1
           1444.03 5066.77966  5066.7773   4.0 1 1
           1458.76 5066.13623  5066.1355   4.0 1 1
           1493.74 5064.60369   5064.602   6.0 1 1
           1518.46 5063.51693  5063.5157   4.0 1 1
           1531.75 5062.93097  5062.9325   4.0 1 1
           1551.96 5062.03813  5062.0371   4.0 1 1
           1560.53 5061.65909  5061.6562   4.0 1 1
           1601.01 5059.86252  5059.8611   6.0 1 1
```

- real/fitted wavelength

- marked wavelength

11) Trim Object

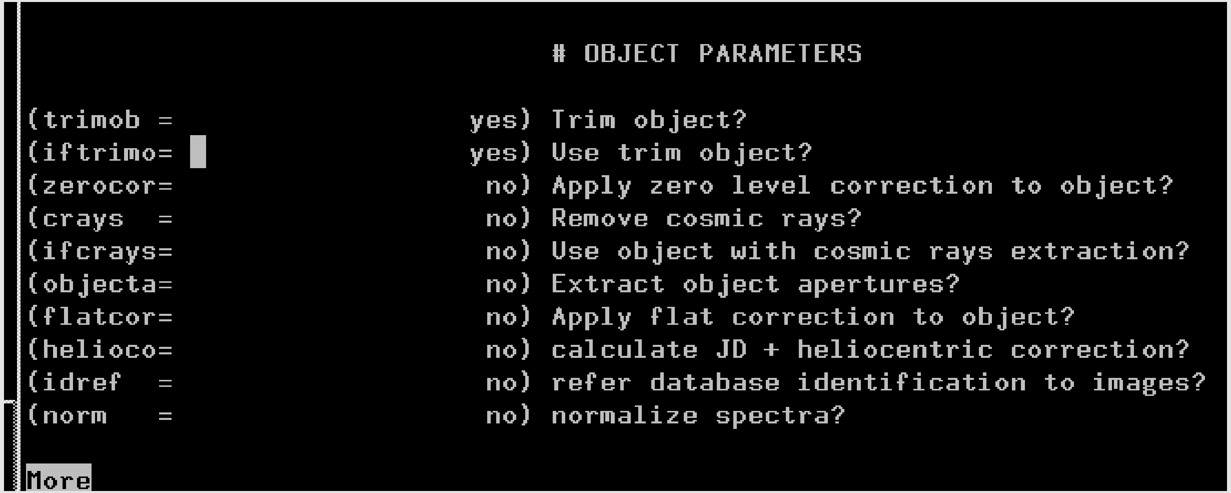
IMPORTANT, first check
these parameters are ok!



```
                                   I R A F
                    Image Reduction and Analysis Facility
PACKAGE = clpackage
   TASK = oesred

input   =   e202306290034.fit  Spectrum target to reduce(.fit)
(output =              alplyr) Output filename
(idtarge=             alp Lyr) Target name on header
(napertu=                  49) Number of apertures to be found
(id     =                0022) Observation id number
```

iftrimc = yes
iddatab = no
idfolder = idcomp
idencom = no

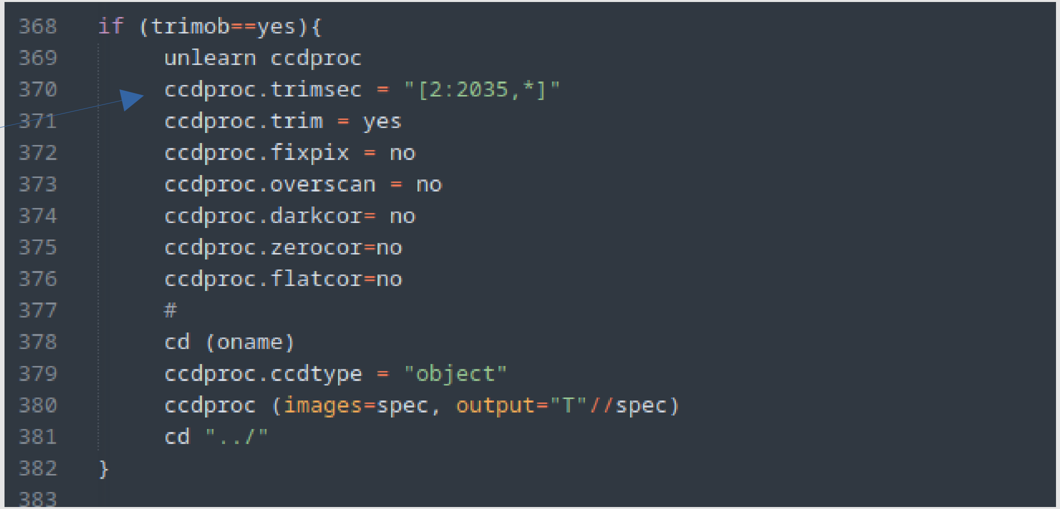
trimob = yes
iftrimo = yes
:go



```
                                   # OBJECT PARAMETERS

(trimob =                      yes) Trim object?
(iftrimo=                      yes) Use trim object?
(zerocor=                       no) Apply zero level correction to object?
(crays  =                       no) Remove cosmic rays?
(ifcrays=                       no) Use object with cosmic rays extraction?
(objecta=                       no) Extract object apertures?
(flatcor=                       no) Apply flat correction to object?
(helioco=                       no) calculate JD + heliocentric correction?
(idref  =                       no) refer database identification to images?
(norm   =                       no) normalize spectra?

More
```

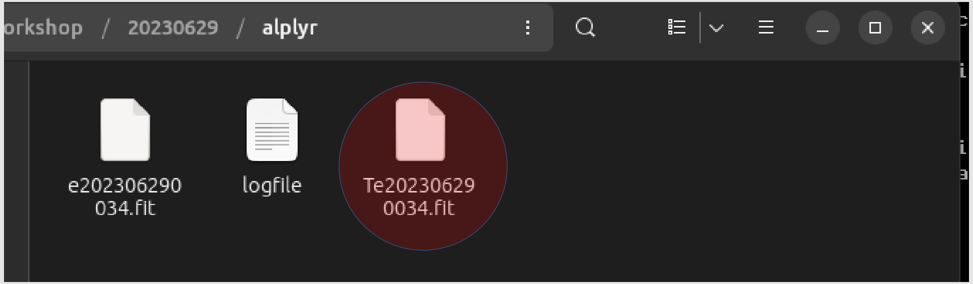
11) Trim Object

- trimsection has to be the same as calibration DON’T change!



```
if (trimob==yes){
    unlearn ccdproc
    ccdproc.trimsec = "[2:2035,*]"
    ccdproc.trim = yes
    ccdproc.fixpix = no
    ccdproc.overscan = no
    ccdproc.darkcor= no
    ccdproc.zerocor=no
    ccdproc.flatcor=no
    #
    cd (oname)
    ccdproc.ccdtype = "object"
    ccdproc (images=spec, output="T"//spec)
    cd "../"
}
```

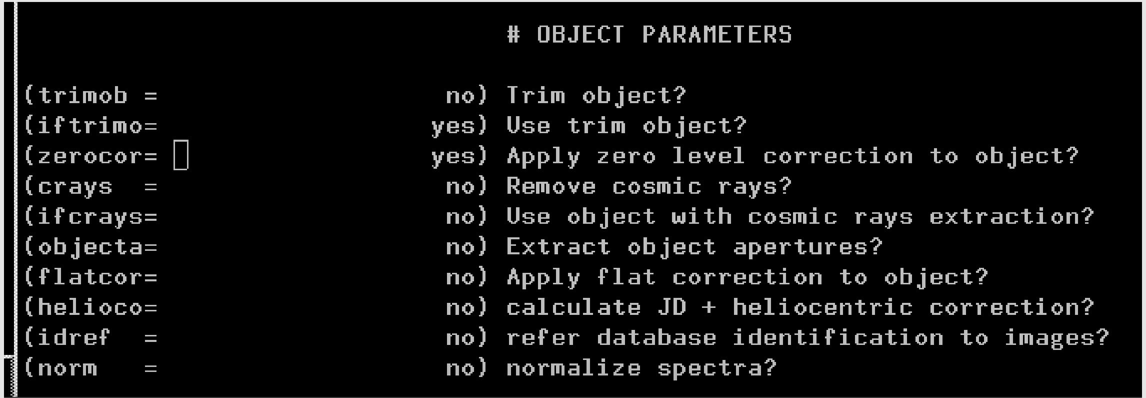
New file: **alplyr/Te2023*.fit**



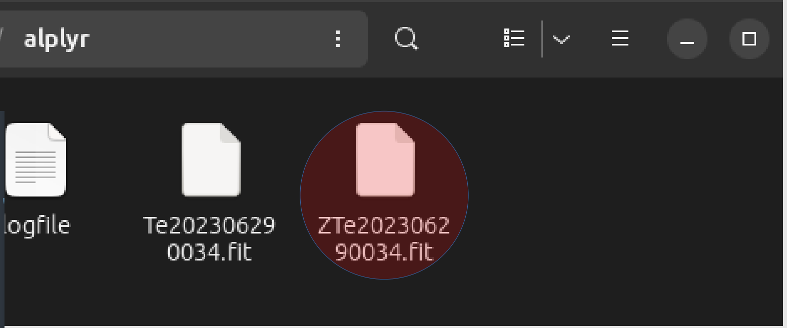
12) Bias correction object

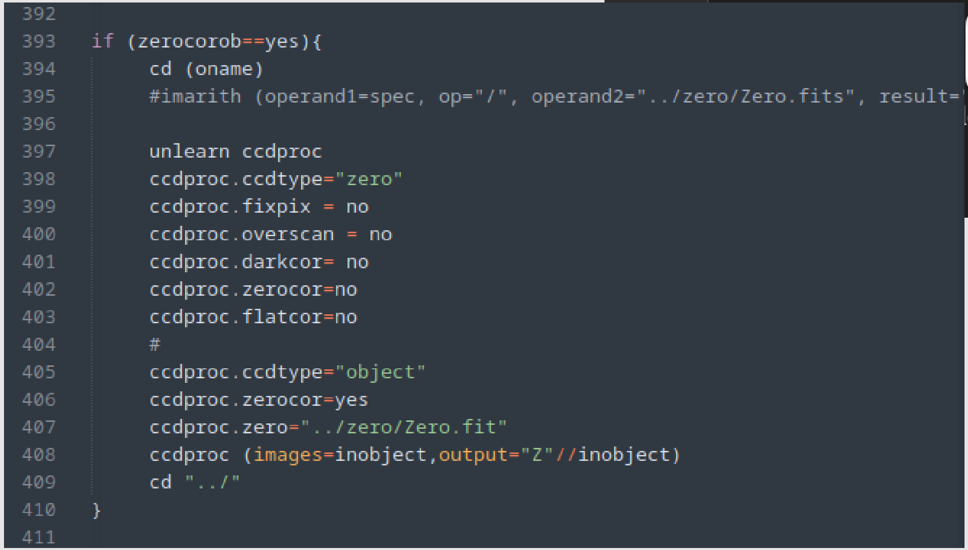
iftrimc = yes
trimob = no
iftrimo = yes
zerocor = yes
:go

New file: **alplyr/ZTe2023*.fit**



```
                                  # OBJECT PARAMETERS

(trimob =                      no) Trim object?
(iftrimo=                     yes) Use trim object?
(zerocor=                     yes) Apply zero level correction to object?
(crays  =                      no) Remove cosmic rays?
(ifcrays=                      no) Use object with cosmic rays extraction?
(objecta=                      no) Extract object apertures?
(flatcor=                      no) Apply flat correction to object?
(helioco=                      no) calculate JD + heliocentric correction?
(idref  =                      no) refer database identification to images?
(norm   =                      no) normalize spectra?
```





```

if (zerocorob==yes){
     cd (oname)
     #imarith (operand1=spec, op="/", operand2="../zero/Zero.fits", result='

     unlearn ccdproc
     ccdproc.ccdtype="zero"
     ccdproc.fixpix = no
     ccdproc.overscan = no
     ccdproc.darkcor= no
     ccdproc.zerocor=no
     ccdproc.flatcor=no
     #
     ccdproc.ccdtype="object"
     ccdproc.zerocor=yes
     ccdproc.zero="../zero/Zero.fit"
     ccdproc (images=inobject,output="Z"//inobject)
     cd "../"
}

```

13) Cosmic Rays – LACOS (*2001PASP..113.1420V*)

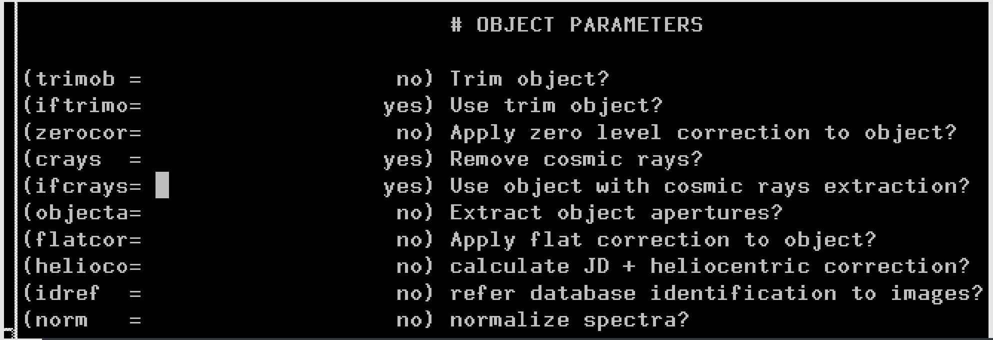
http://www.astro.yale.edu/dokkum/lacosmic/

To remove cosmic rays

iftrimc = yes
iftrimo = yes
zerocor = no
crays = yes
ifcrays = yes
:go

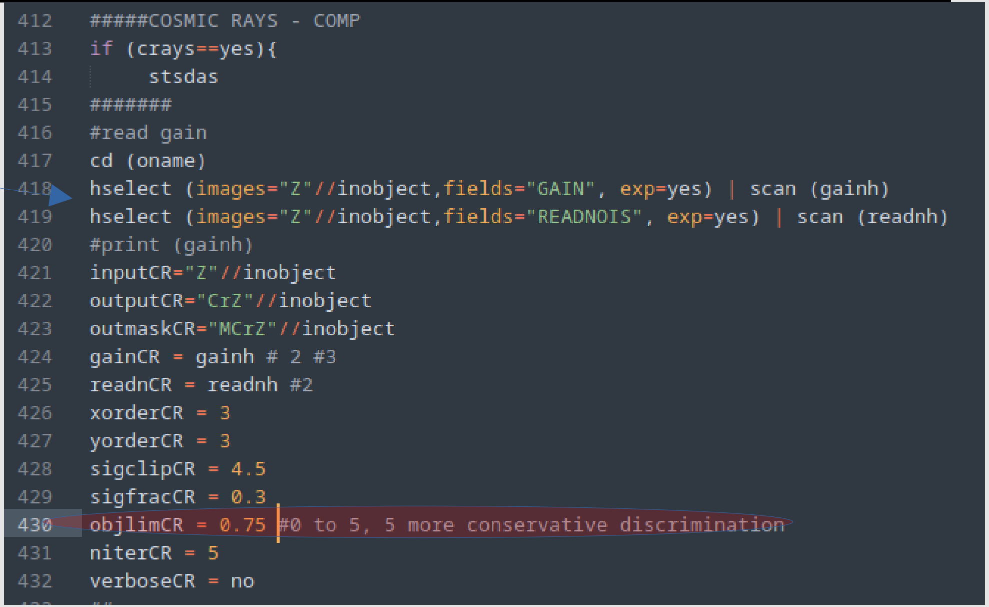
But it's better to skip cosmic removal

iftrimc = yes
iftrimo = yes
zerocor = no
crays = no
ifcrays = no
:q



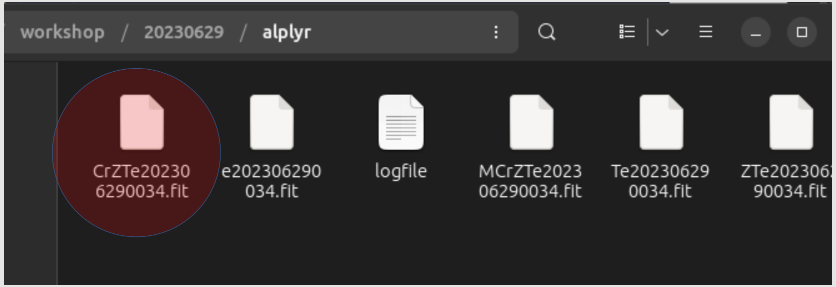
```
                                  # OBJECT PARAMETERS

(trimob =                      no) Trim object?
(iftrimo=                     yes) Use trim object?
(zerocor=                      no) Apply zero level correction to object?
(crays  =                     yes) Remove cosmic rays?
(ifcrays=                     yes) Use object with cosmic rays extraction?
(objecta=                      no) Extract object apertures?
(flatcor=                      no) Apply flat correction to object?
(helioco=                      no) calculate JD + heliocentric correction?
(idref  =                      no) refer database identification to images?
(norm   =                      no) normalize spectra?
```

- read GAIN and READtoNOISE from header



```
#####COSMIC RAYS - COMP
if (crays==yes){
    stsdas
#######
#read gain
cd (oname)
hselect (images="Z"//inobject,fields="GAIN", exp=yes) | scan (gainh)
hselect (images="Z"//inobject,fields="READNOIS", exp=yes) | scan (readnh)
#print (gainh)
inputCR="Z"//inobject
outputCR="CrZ"//inobject
outmaskCR="MCrZ"//inobject
gainCR = gainh # 2 #3
readnCR = readnh #2
xorderCR = 3
yorderCR = 3
sigclipCR = 4.5
sigfracCR = 0.3
objlimCR = 0.75 #0 to 5, 5 more conservative discrimination
niterCR = 5
verboseCR = no
```

New files: **alplyr/CrZTe2023*.fit** ← cleaned
alplyr/MCrZTe2023*.fit ← mask



14) Extract apertures - Object

- Template: AZTflat.fit

iftrimc = yes

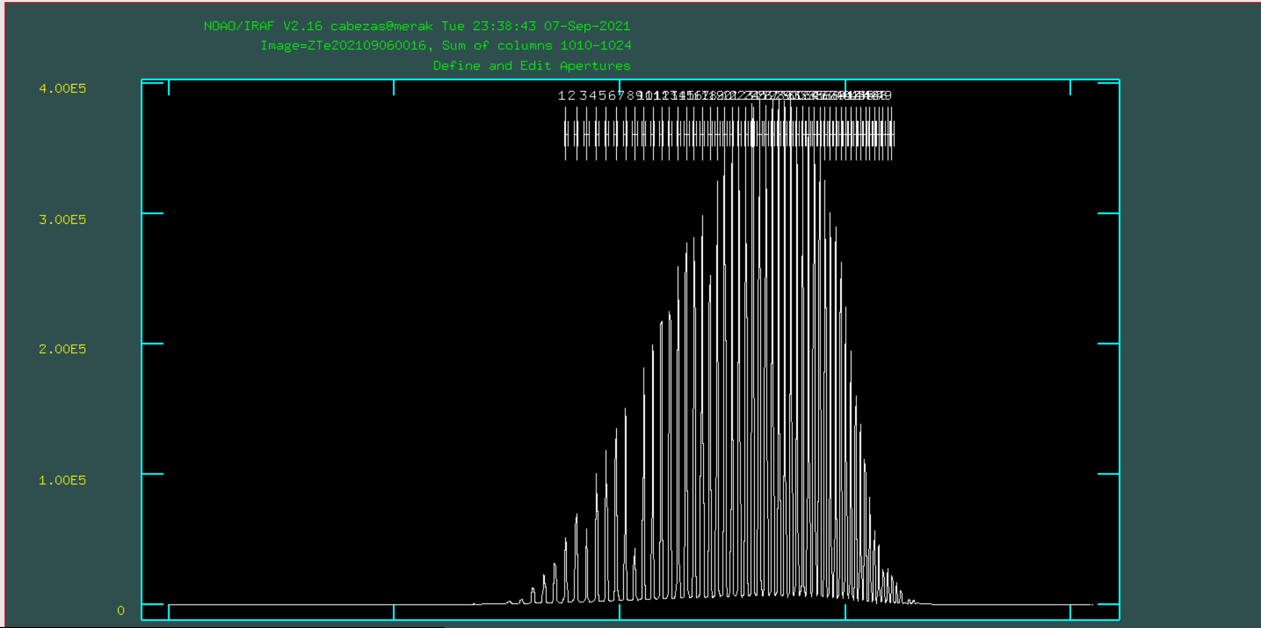
iftrimo = yes

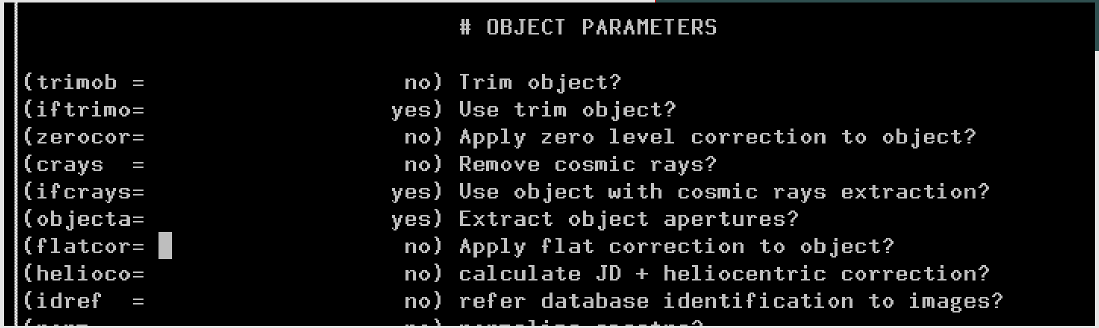
crays = no

ifcrays = no (or yes)

objecta = yes

:go





```
                         # OBJECT PARAMETERS

(trimob =                      no) Trim object?
(iftrimo=                     yes) Use trim object?
(zerocor=                      no) Apply zero level correction to object?
(crays  =                      no) Remove cosmic rays?
(ifcrays=                     yes) Use object with cosmic rays extraction?
(objecta=                     yes) Extract object apertures?
(flatcor=                      no) Apply flat correction to object?
(helioco=                      no) calculate JD + heliocentric correction?
(idref  =                      no) refer database identification to images?
(norm   =                      no) normalize spectra?
```

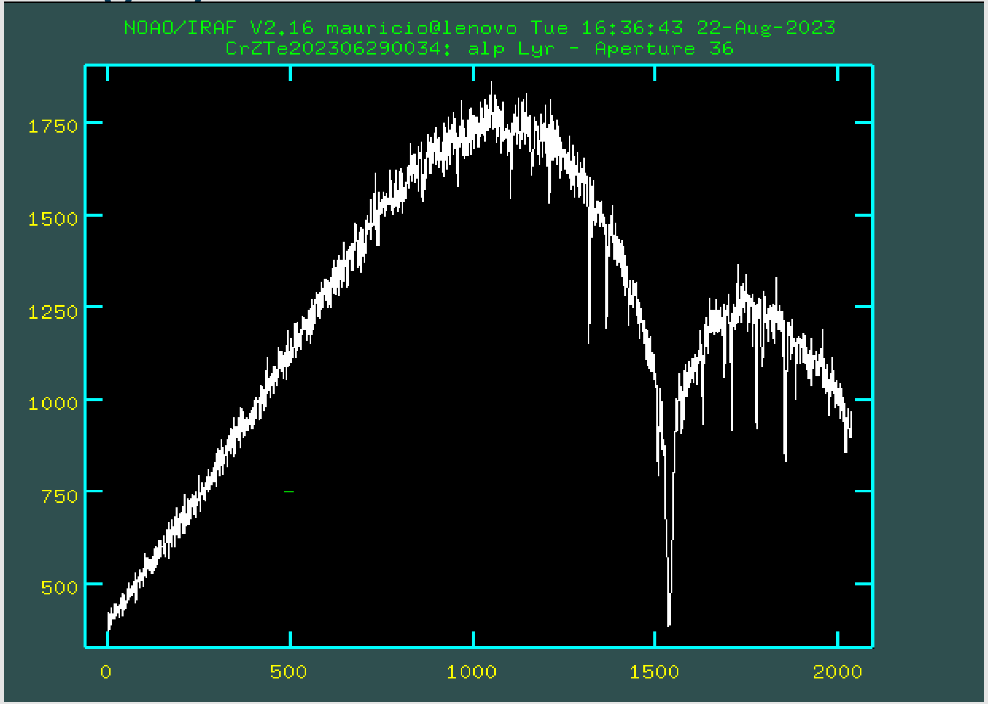
14) Extract apertures - Object

Edit apertures for CrZTe2023*? (yes):
Review extracted spectrum for aperture 1 from CrZTe2023*? (yes)
Write apertures for CrZTe2023* to database? (yes):
q → if yes, with q you move to the next aperture
Extract aperture spectra for CrZTe2023*? (yes):

First view of our target!!!
BUT!! :)

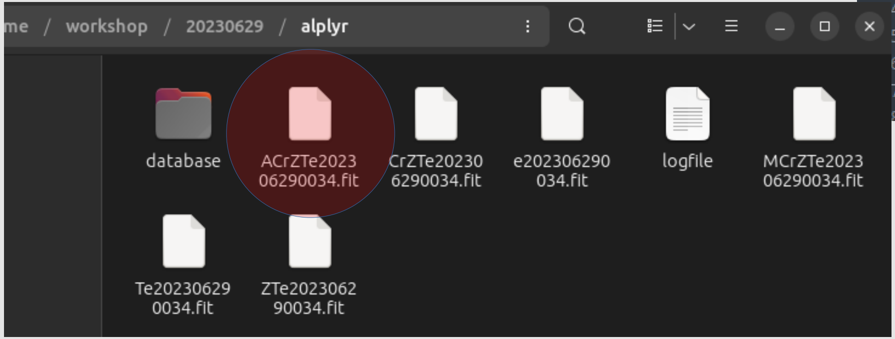
Still in pixels :(

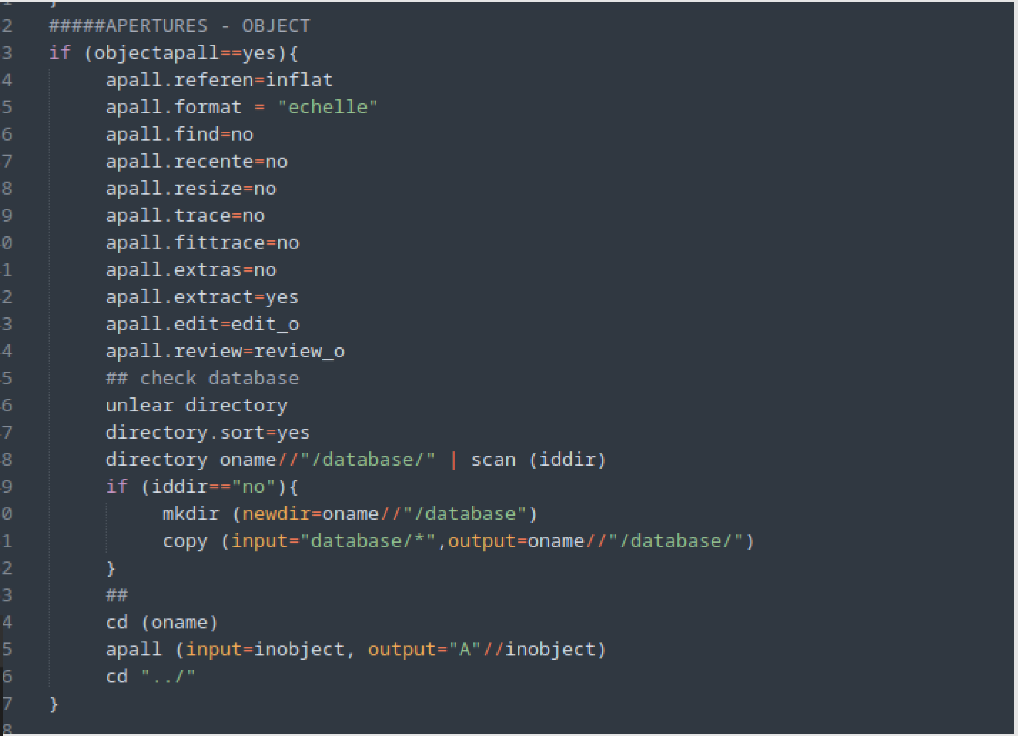
For example aperture 36 is
the spectral region around H-alpha



14) Extract apertures - Object

new file: **alplyr/ACrZTe2023*.fit**





```
#####APERTURES - OBJECT
if (objectapall==yes){
    apall.referen=inflat
    apall.format = "echelle"
    apall.find=no
    apall.recente=no
    apall.resize=no
    apall.trace=no
    apall.fittrace=no
    apall.extras=no
    apall.extract=yes
    apall.edit=edit_o
    apall.review=review_o
    ## check database
    unlear directory
    directory.sort=yes
    directory oname//"/database/" | scan (iddir)
    if (iddir=="no"){
        mkdir (newdir=oname//"/database")
        copy (input="database/*",output=oname//"/database/")
    }
    ##
    cd (oname)
    apall (input=inobject, output="A"//inobject)
    cd "../"
}
```

15) Flat correction

iftrimc = yes

iftrimo = yes

ifcrays = yes/no

objecta = no

flatcor = yes

:go

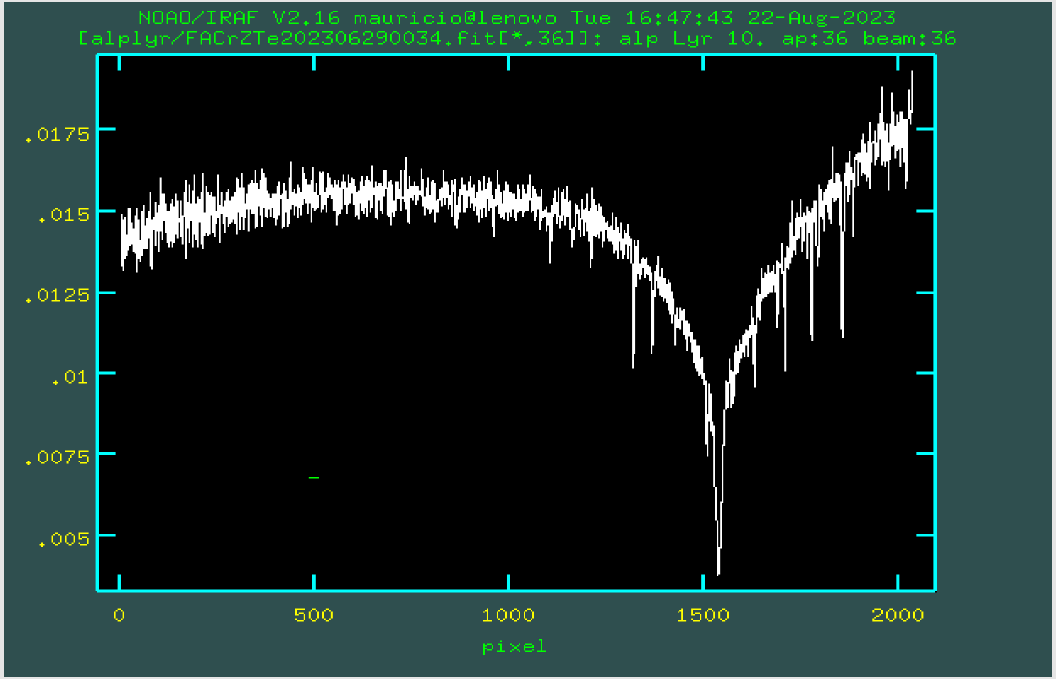
To see or plot use the task

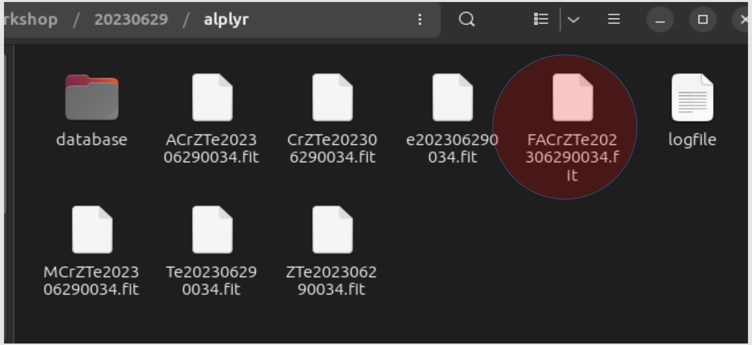
splot spec.fit

To move between orders

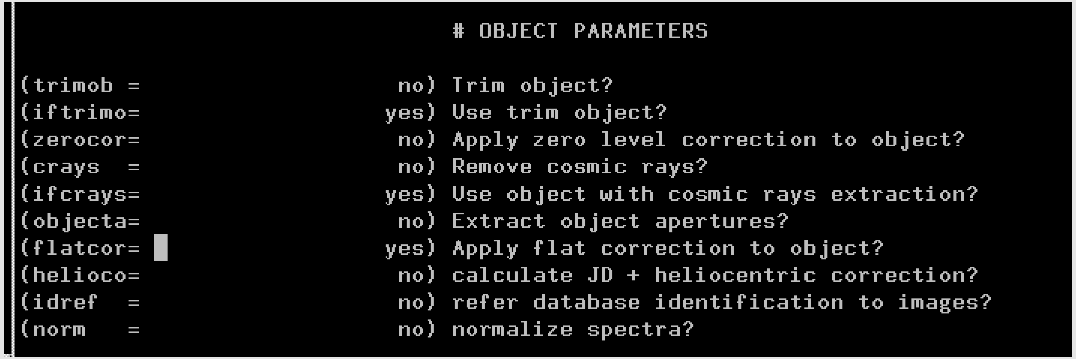
(→ to the right/ higher aperture

) → to the left/ lower aperture





new file: **FACrZTe2023*.fit**



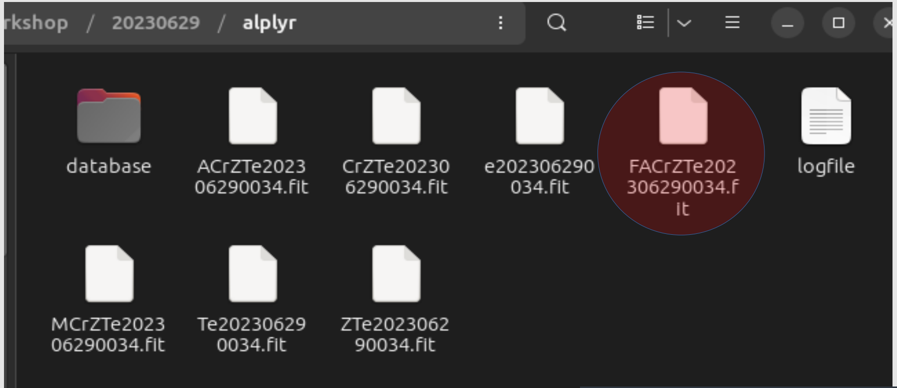
```
                        # OBJECT PARAMETERS

(trimob =                   no) Trim object?
(iftrimo=                  yes) Use trim object?
(zerocor=                   no) Apply zero level correction to object?
(crays  =                   no) Remove cosmic rays?
(ifcrays=                  yes) Use object with cosmic rays extraction?
(objecta=                   no) Extract object apertures?
(flatcor=                  yes) Apply flat correction to object?
(helioco=                   no) calculate JD + heliocentric correction?
(idref  =                   no) refer database identification to images?
(norm   =                   no) normalize spectra?
```

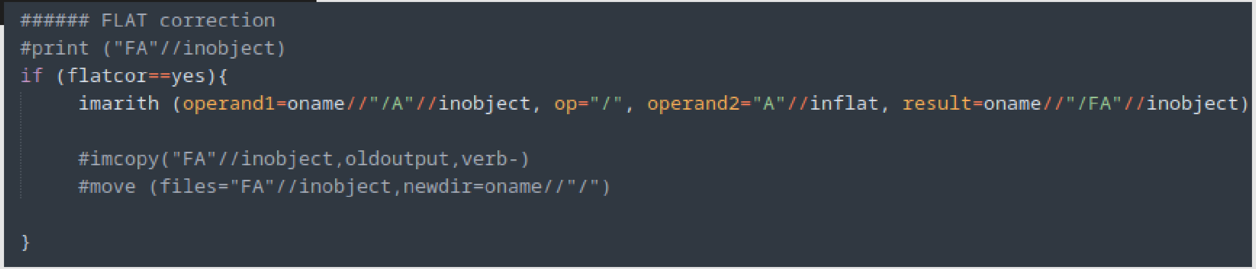
15) Flat correction

In Flat correction we have to divide. By dividing the science spectrum by the flat-field image, we are effectively correcting for the variations in

- sensitivity
- illumination
- detector response.

new file: **FACrZTe2023*.fit**

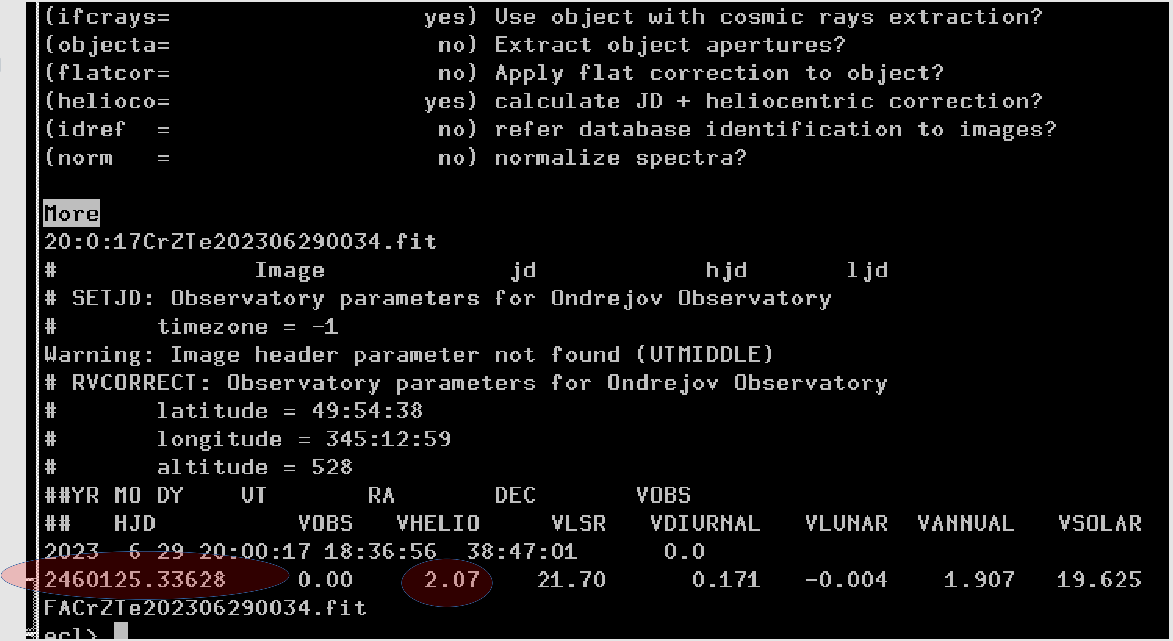




```
###### FLAT correction
#print ("FA"//inobject)
if (flatcor==yes){
    imarith (operand1=oname//"/A"//inobject, op="/", operand2="A"//inflat, result=oname//"/FA"//inobject)

    #imcopy("FA"//inobject,oldoutput,verb-)
    #move (files="FA"//inobject,newdir=oname//"/")

}
```

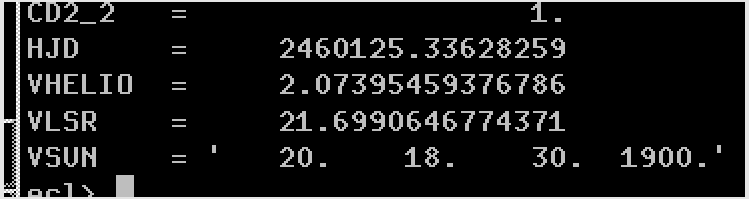
16) JD + heliocentric correction

iftrimc = yes
iftrimo = yes
ifcrays = no
flatcor = no
helioco = yes
:go



```
(ifcrays=                      yes) Use object with cosmic rays extraction?
(objecta=                       no) Extract object apertures?
(flatcor=                       no) Apply flat correction to object?
(helioco=                      yes) calculate JD + heliocentric correction?
(idref  =                       no) refer database identification to images?
(norm   =                       no) normalize spectra?

More
20:0:17CrZTe202306290034.fit
#              Image            jd            hjd       ljd
# SETJD: Observatory parameters for Ondrejov Observatory
#       timezone = -1
Warning: Image header parameter not found (UTMIDDLE)
# RVCORRECT: Observatory parameters for Ondrejov Observatory
#       latitude = 49:54:38
#       longitude = 345:12:59
#       altitude = 528
##YR MO DY    UT        RA        DEC       VOBS
##   HJD          VOBS   VHELIO     VLSR   VDIURNAL    VLUNAR  VANNUAL    VSOLAR
2023  6 29 20:00:17 18:36:56  38:47:01      0.0
2460125.33628     0.00     2.07     21.70      0.171    -0.004     1.907   19.625
FACrZTe202306290034.fit
ecl>
```

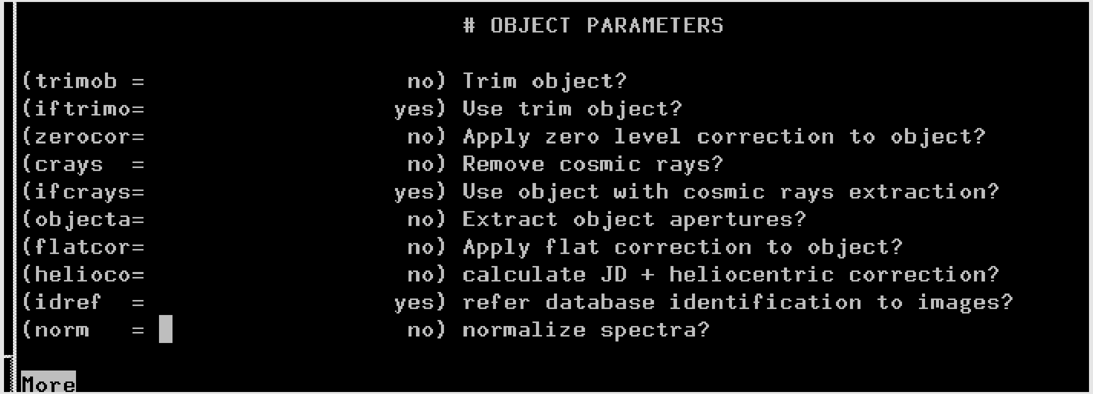
Check long header:
imhead FCrAZTe2023*.fit l+



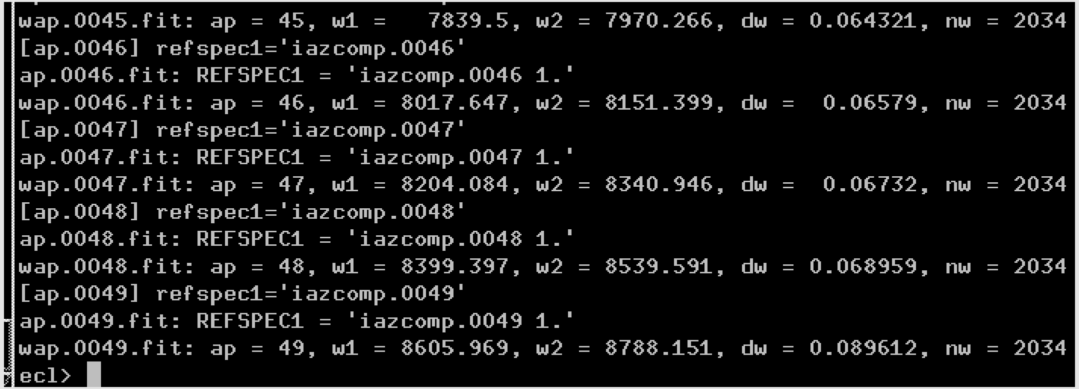
```
CD2_2   =                    1.
HJD     =      2460125.33628259
VHELIO  =      2.07395459376786
VLSR    =      21.6990646774371
VSUN    = '    20.     18.      30.   1900.'
ecl>
```

17) Ref spectrum

iftrimc = yes
iftrimo = yes
ifcrays = no
helioco = no
idref = yes
:go



```
                                # OBJECT PARAMETERS

(trimob =                       no) Trim object?
(iftrimo=                      yes) Use trim object?
(zerocor=                       no) Apply zero level correction to object?
(crays  =                       no) Remove cosmic rays?
(ifcrays=                      yes) Use object with cosmic rays extraction?
(objecta=                       no) Extract object apertures?
(flatcor=                       no) Apply flat correction to object?
(helioco=                       no) calculate JD + heliocentric correction?
(idref  =                      yes) refer database identification to images?
(norm   =                       no) normalize spectra?

More
```



```
wap.0045.fit: ap = 45, w1 =   7839.5, w2 = 7970.266, dw = 0.064321, nw = 2034
[ap.0046] refspec1='iazcomp.0046'
ap.0046.fit: REFSPEC1 = 'iazcomp.0046 1.'
wap.0046.fit: ap = 46, w1 = 8017.647, w2 = 8151.399, dw =  0.06579, nw = 2034
[ap.0047] refspec1='iazcomp.0047'
ap.0047.fit: REFSPEC1 = 'iazcomp.0047 1.'
wap.0047.fit: ap = 47, w1 = 8204.084, w2 = 8340.946, dw =  0.06732, nw = 2034
[ap.0048] refspec1='iazcomp.0048'
ap.0048.fit: REFSPEC1 = 'iazcomp.0048 1.'
wap.0048.fit: ap = 48, w1 = 8399.397, w2 = 8539.591, dw = 0.068959, nw = 2034
[ap.0049] refspec1='iazcomp.0049'
ap.0049.fit: REFSPEC1 = 'iazcomp.0049 1.'
wap.0049.fit: ap = 49, w1 = 8605.969, w2 = 8788.151, dw = 0.089612, nw = 2034
ecl>
```

17) Ref spectrum

Read database of wavelength calibration and apply!

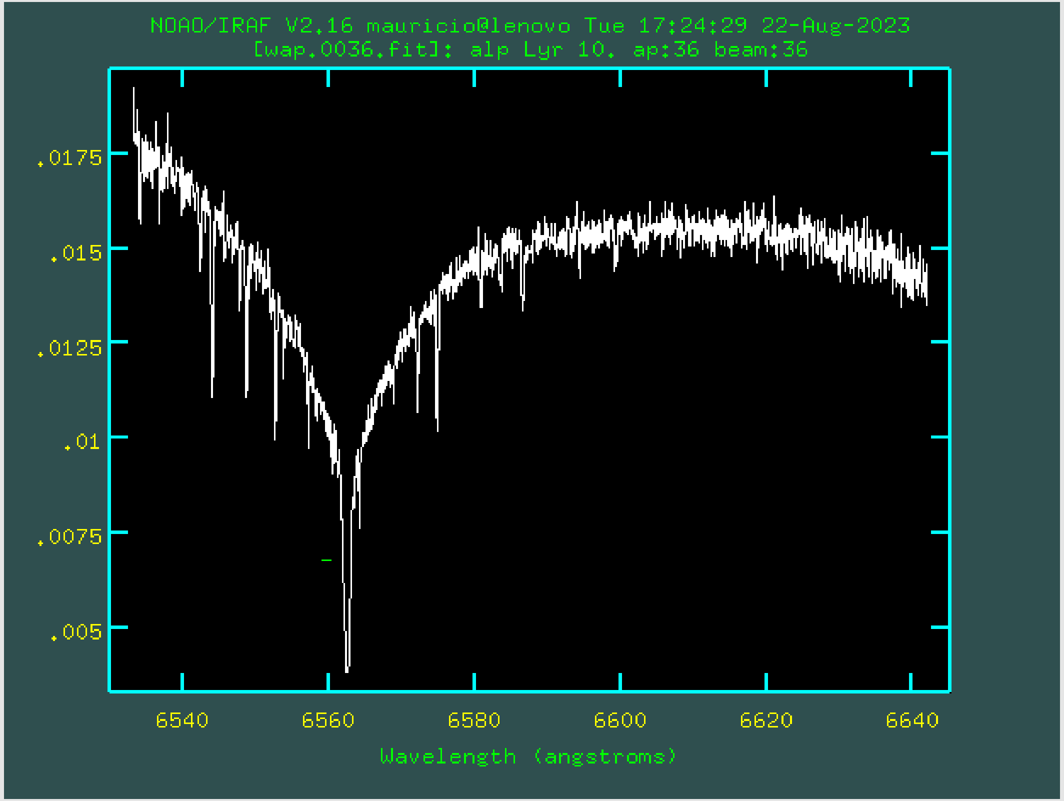
splot wap.0036.fit

NO more pixels!!! :)

→ aperture 36 → H-alpha

Check if ok

Good Wavelength calibration !!



NOAO/IRAF V2.16 mauricio@lenovo Tue 17:24:29 22-Aug-2023
[wap.0036.fit]: alp Lyr 10. ap:36 beam:36
.0175
.015
.0125
.01
.0075
.005
6540
6560
6580
6600
6620
6640
Wavelength (angstroms)

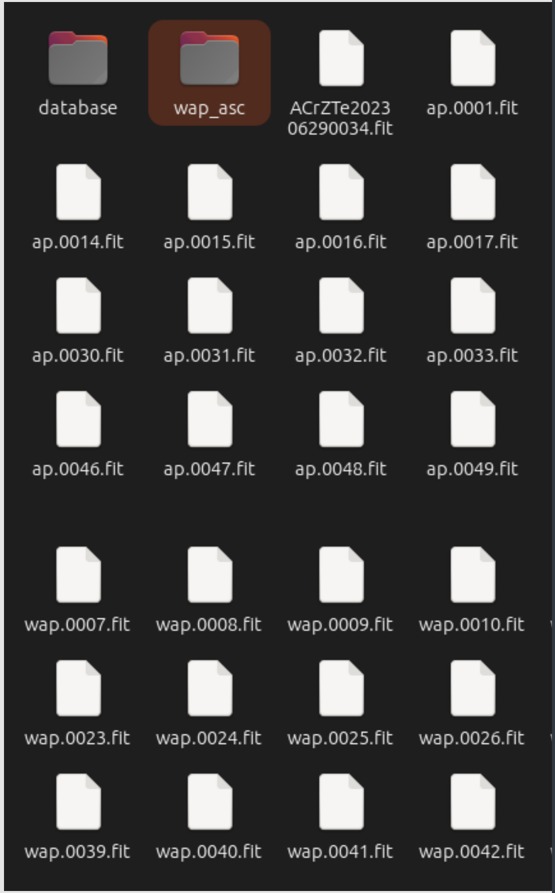
17) Ref spectrum

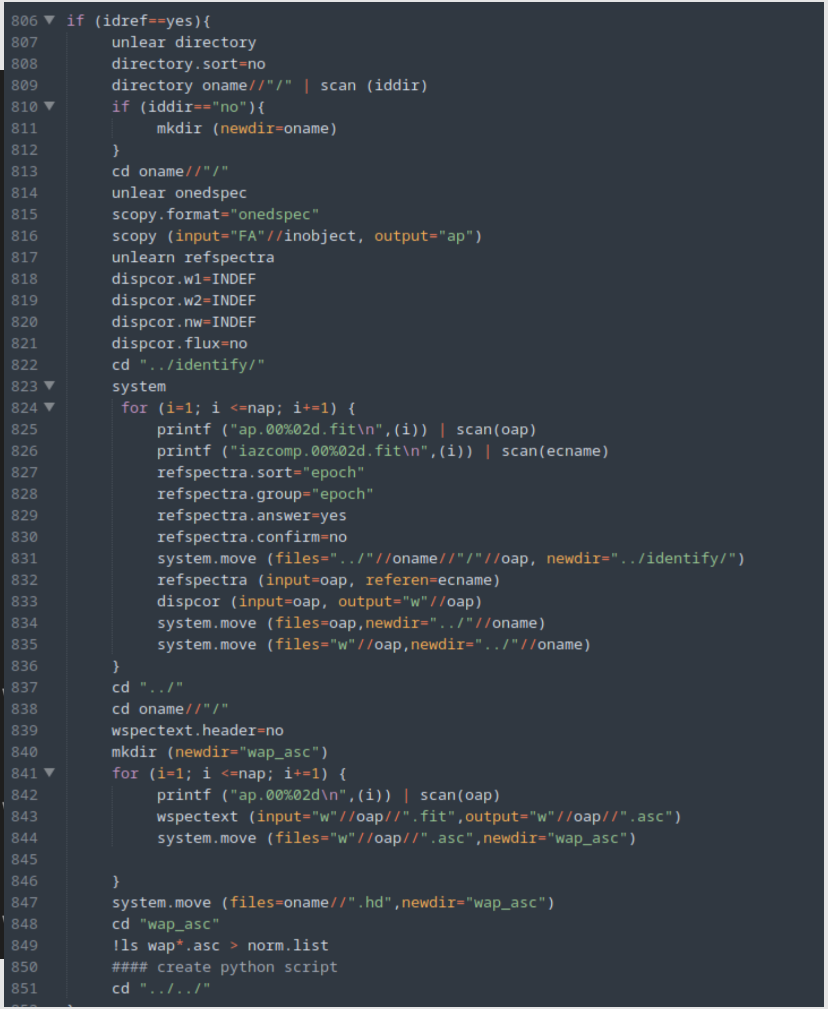
New files:

ap.00XX.fit –
1D spectra in pixels

wap.00XX.fit –
1D spectra in Angstrom

wap_asc/wap.00XX.asc –
ascii file for each aperture (useful)





```
if (idref==yes){
    unlear directory
    directory.sort=no
    directory oname//"/" | scan (iddir)
    if (iddir=="no"){
        mkdir (newdir=oname)
    }
    cd oname//"/"
    unlear onedspec
    scopy.format="onedspec"
    scopy (input="FA"//inobject, output="ap")
    unlearn refspectra
    dispcor.w1=INDEF
    dispcor.w2=INDEF
    dispcor.nw=INDEF
    dispcor.flux=no
    cd "../identify/"
    system
     for (i=1; i <=nap; i+=1) {
        printf ("ap.00%02d.fit\n",(i)) | scan(oap)
        printf ("iazcomp.00%02d.fit\n",(i)) | scan(ecname)
        refspectra.sort="epoch"
        refspectra.group="epoch"
        refspectra.answer=yes
        refspectra.confirm=no
        system.move (files="../"//oname//"/"//oap, newdir="../identify/")
        refspectra (input=oap, referen=ecname)
        dispcor (input=oap, output="w"//oap)
        system.move (files=oap,newdir="../"//oname)
        system.move (files="w"//oap,newdir="../"//oname)
    }
    cd "../"
    cd oname//"/"
    wspectext.header=no
    mkdir (newdir="wap_asc")
    for (i=1; i <=nap; i+=1) {
        printf ("ap.00%02d\n",(i)) | scan(oap)
        wspectext (input="w"//oap//".fit",output="w"//oap//".asc")
        system.move (files="w"//oap//".asc",newdir="wap_asc")

    }
    system.move (files=oname//".hd",newdir="wap_asc")
    cd "wap_asc"
    !ls wap*.asc > norm.list
    #### create python script
    cd "../../"
```

18) Normalization

iftrimc = yes

iftrimo = yes

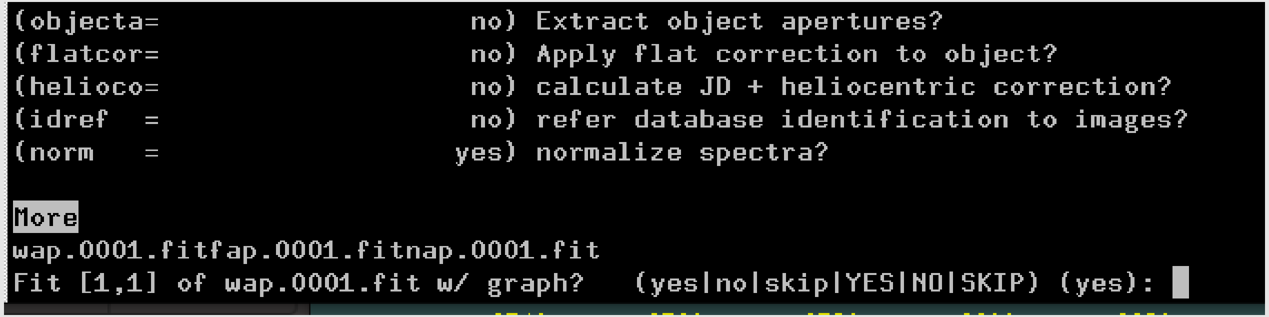
ifcrays = no

idref = no

norm = yes

:go

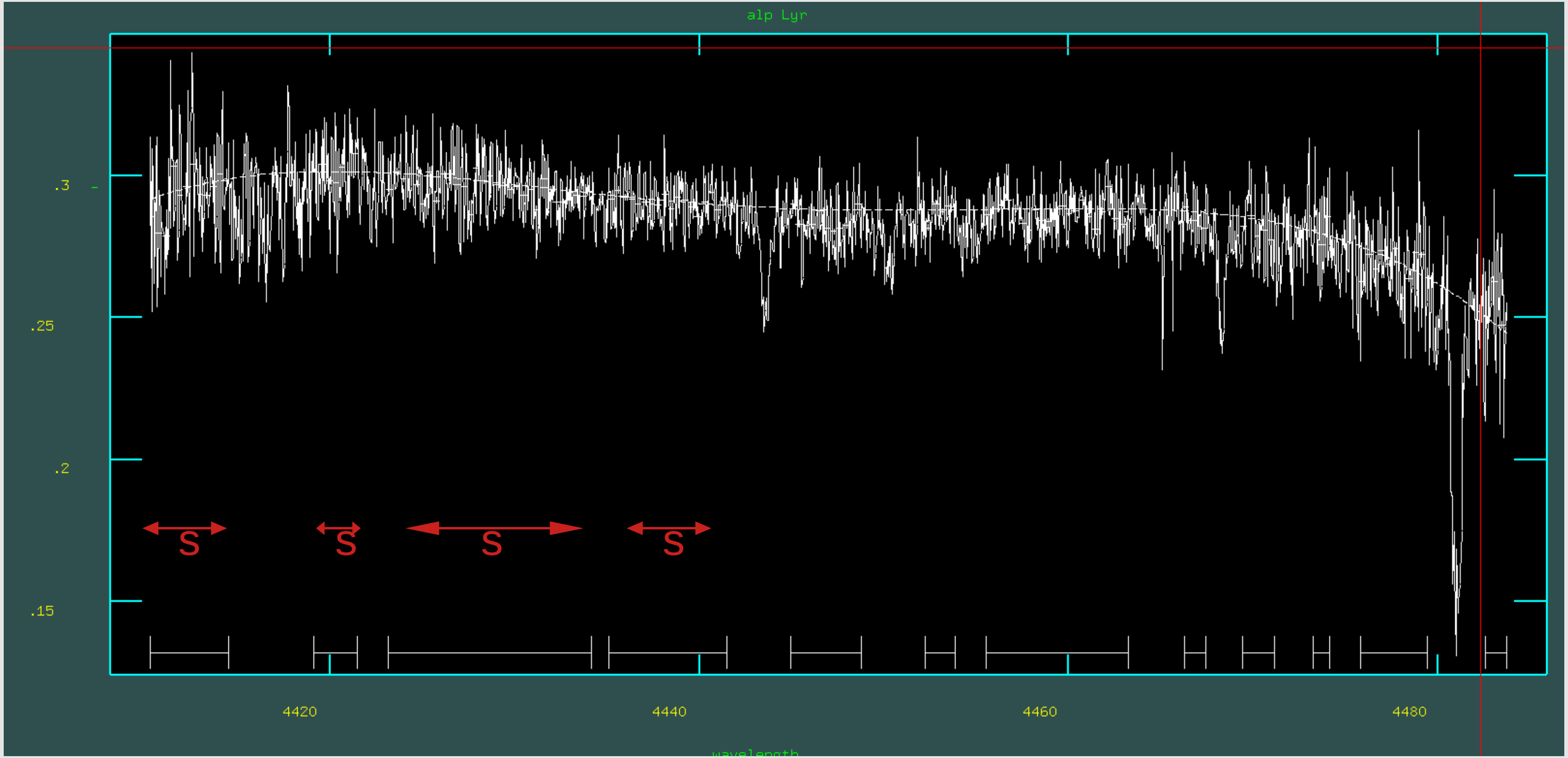
Fit [1,1] of wap.0001.fit w/ graph? (yes|no|skip|YES|NO|SKIP) (yes):



```
(objecta=                           no) Extract object apertures?
(flatcor=                           no) Apply flat correction to object?
(helioco=                           no) calculate JD + heliocentric correction?
(idref  =                           no) refer database identification to images?
(norm   =                          yes) normalize spectra?

More
wap.0001.fitfap.0001.fitnap.0001.fit
Fit [1,1] of wap.0001.fit w/ graph?   (yes|no|skip|YES|NO|SKIP) (yes):
```

18) Normalization

- change order **:o N**
- residuals **j**
- high rejection **:hi N**
- low rejection **:lo N**
- function **:f legendre/spline3/chebyshev**
- select region **s..s**
- delete region **z**
- delete all regions **t**
- zoom/window **w, e..e**
  - resize **w,a**
  - move right **w,r**
  - move left **w,l**
  - move up **w,u**
  - move down **w,d**

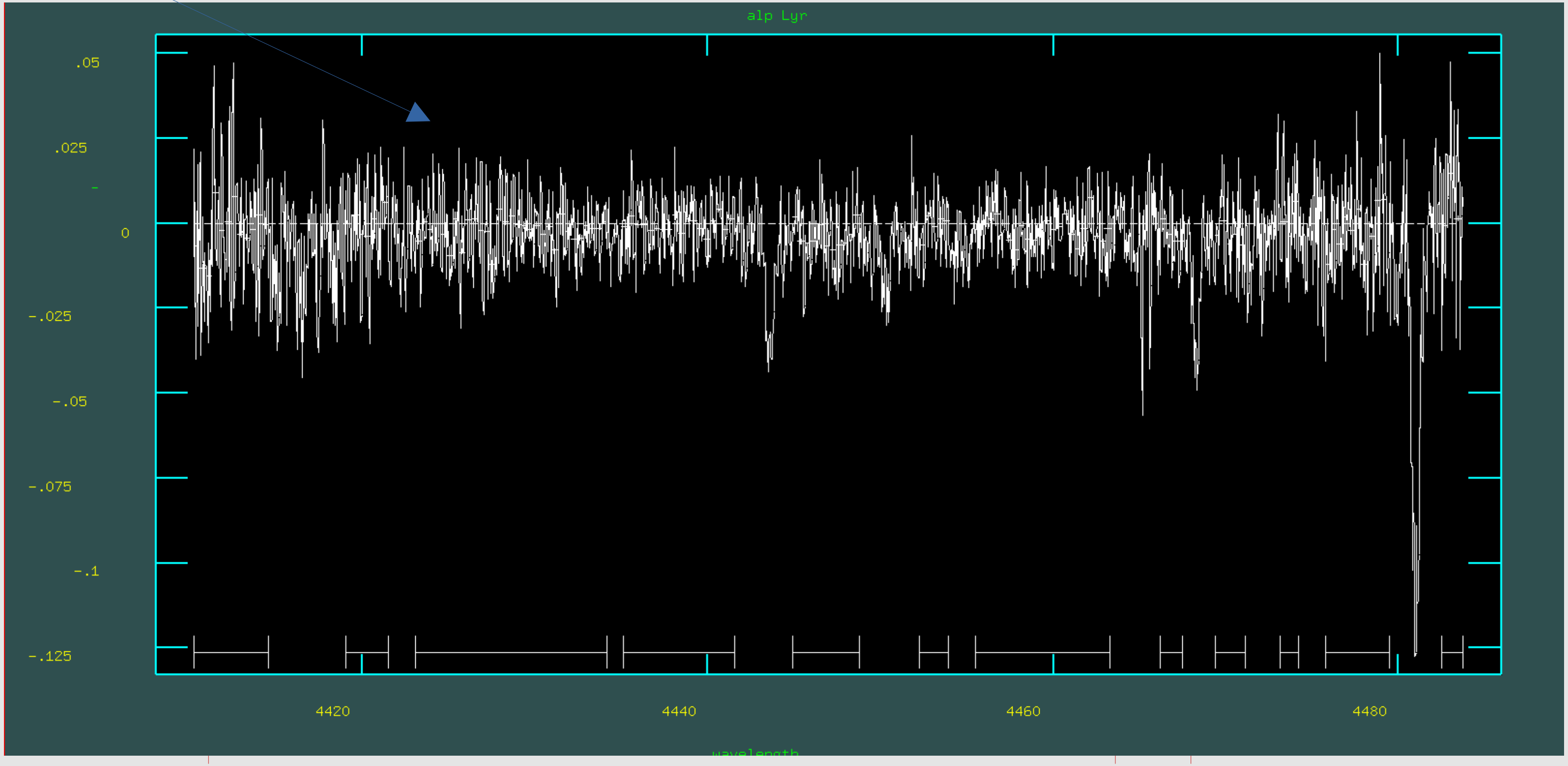


18) Normalization

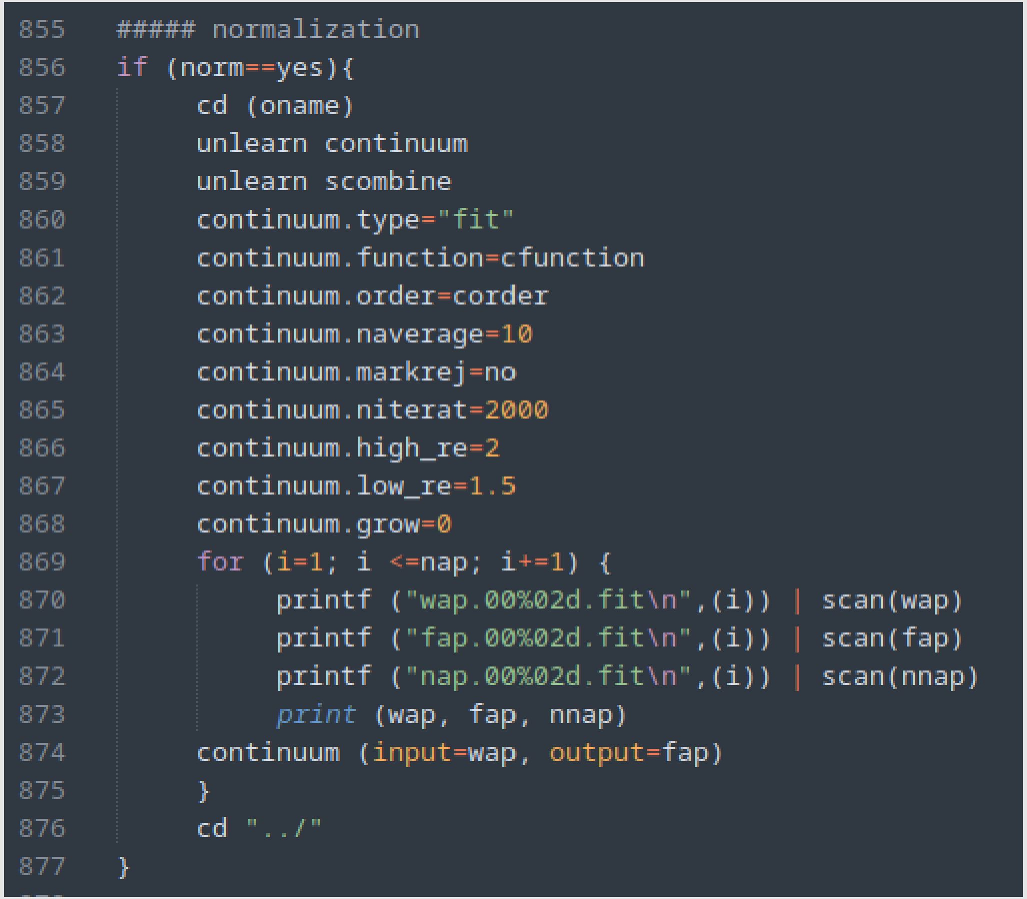
- change order **:o N**
- residuals **j**
- high rejection **:hi N**

- low rejection **:lo N**
- function **:f legendre/spline3/chebyshev**
- select region **s..s**
- delete region **z**
- delete all regions **t**

- zoom/window **w, e..e**
  resize **w,a**
  move right **w,r**
  move left **w,l**
  move up **w,u**
  move down **w,d**



18) Normalization



```
##### normalization
if (norm==yes){
    cd (oname)
    unlearn continuum
    unlearn scombine
    continuum.type="fit"
    continuum.function=cfunction
    continuum.order=corder
    continuum.naverage=10
    continuum.markrej=no
    continuum.niterat=2000
    continuum.high_re=2
    continuum.low_re=1.5
    continuum.grow=0
    for (i=1; i <=nap; i+=1) {
        printf ("wap.00%02d.fit\n",(i)) | scan(wap)
        printf ("fap.00%02d.fit\n",(i)) | scan(fap)
        printf ("nap.00%02d.fit\n",(i)) | scan(nnap)
        print (wap, fap, nnap)
    continuum (input=wap, output=fap)
    }
    cd "../"
}
```

Advises / Good practices

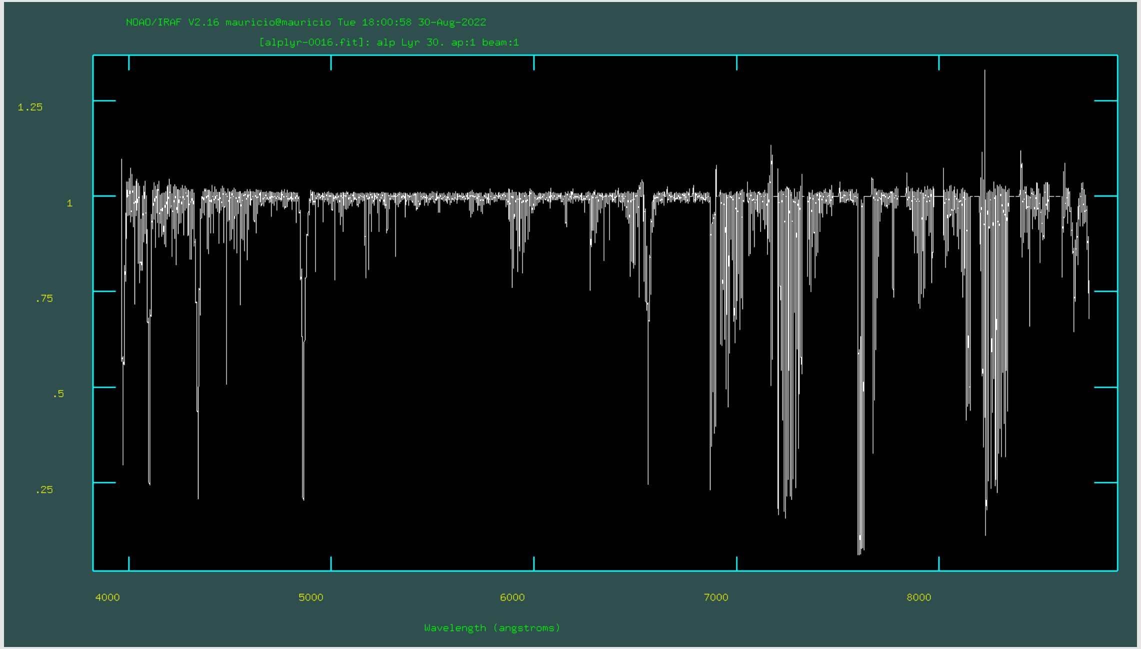
- Start with low order
- Hot stars or wide lines →
  legendre or chebyshev
- sometimes faster just change high/low rejection
- Don’t frustrate normalization is Just experience, there is not ABSOLUTE way

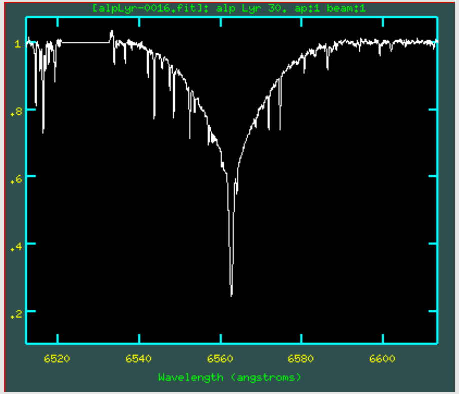
fap.00XX.fit –
Normalised 1D spectra in pixels

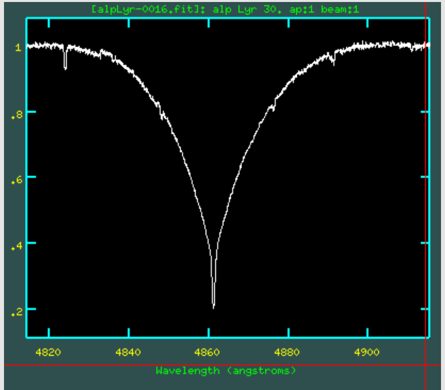
19) merging

```
iftrimc = yes
iftrimo = yes
ifcrays = no
norm = no
ncombine = yes
nrvcorr = yes
:go
```

new file: `DCN-alyr_20210906.fit`







SPLOT

splot alpLyr-0016.fit

Fit: gaussian: **k..k(**or **g)**

lorentzian: **k..l**

voigt: **k..v**

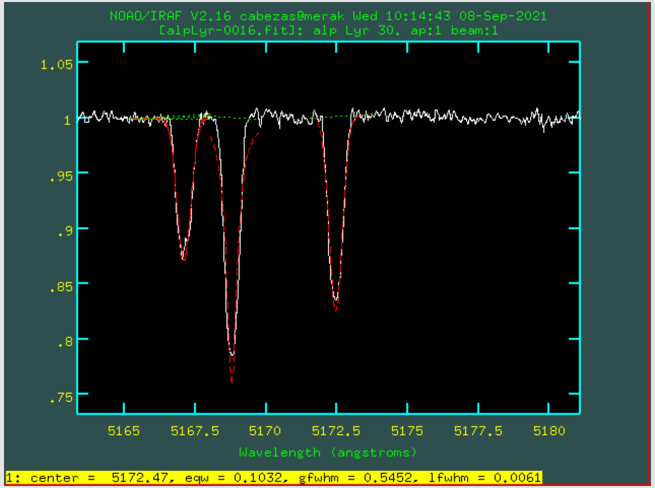
centroid **e..e**

snr: **m..m**

Change unit (angstrom to km/s)

:u km/s 6562.8 an

:u an


```
NOAO/IRAF V2.16 cabezas@merak Wed 10:14:43 08-Sep-2021
[alpLyr-0016.fit]: alp Lyr 30. ap:1 beam:1
```

```
1: center =  5172.47, eqw = 0.1032, gfwhm = 0.5452, lfwhm = 0.0061
```

SPECPLOT

specplot spec1.fit,spec2.fits

Change step: **step 1** (or any number)

See wavelength: **u**

snr: **m..m**

Change unit (angstrom to km/s)

:u km/s 6562.8 an

:u an

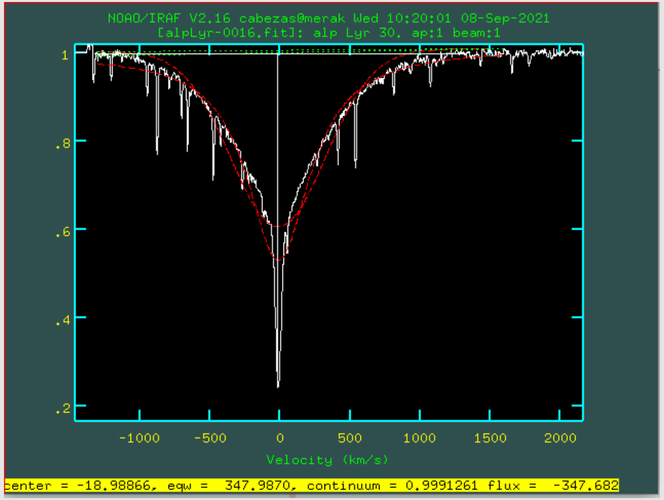
Replot: **r**

Save spectrum as plain text

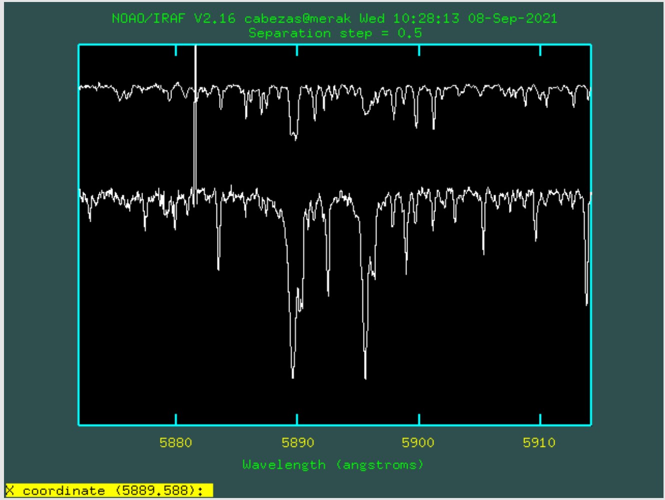
```
onedspec
wspectext spec.fits spec.txt
```

Proper file example:

```
DCN-alyr_20210906.fit
```


```
NOAO/IRAF V2.16 cabezas@merak Wed 10:20:01 08-Sep-2021
[alpLyr-0016.fit]: alp Lyr 30. ap:1 beam:1
```

```
center = -18.98866, eqw =  347.9870, continuum = 0.9991261 flux =  -347.682
```


```
NOAO/IRAF V2.16 cabezas@merak Wed 10:28:13 08-Sep-2021
Separation step = 0.5
```

```
X coordinate (5889.588):
```

General Remarks

- The optimal reduction process always will be different for each instrument.
- IRAF "sometimes" is a bit tricky, but really useful.
- Quick check/inspection of spectra!!
- Versatile program because many parameters (sometimes too many).
- Pre-defined task.
- "opensource" you can write your own task/package.
- Xgterm – nice interactive tool.
- Slow with computation, python/idl/fortran would be good option.
- **IF CRASH!** Check your last parameters (**lpar oesred),** check the new Created files probably you want delete them and run it again. Check the location where you are when you **:go**
- Don't gave up and enjoy (?) :)

Thank you!!

IRAF - useful commands

https://iraf.net

help task

Plot spectrum

splot spec.fit

Plot set of spectra

specplot @spec.list

specplot e*.fit,01.fit,02.fit...

Check header

imhead spec.fit/@spec.list l+ | page

Select some field from header

hselect spec.fit/@spec.list $l,obj-name,exptime yes

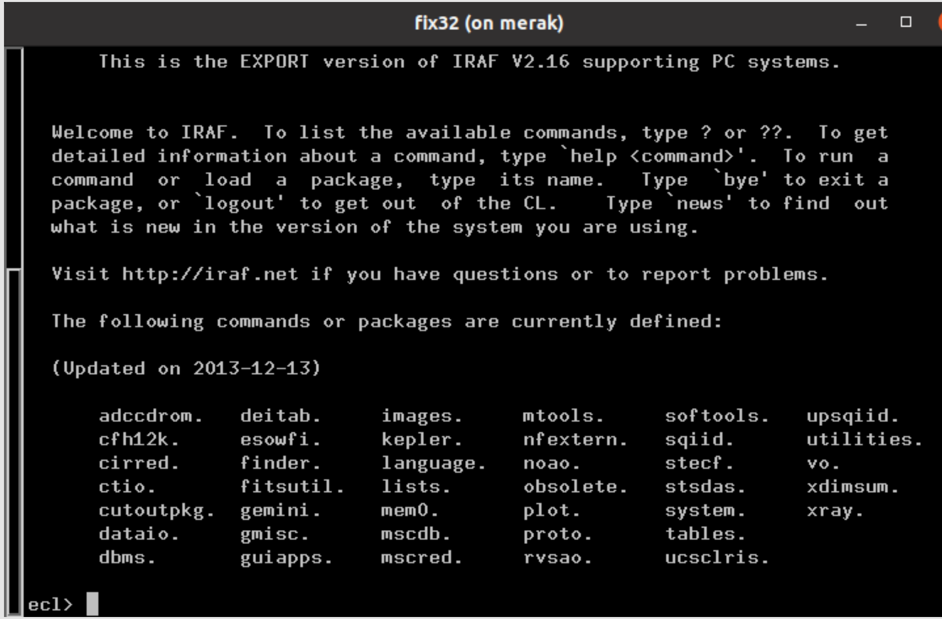
Check stats of spectrum

imstat spec.fit/@spec.list

See image with ds9

!ds9 &

display spec.fit Nframe (nframe=1,..,12)



```
    This is the EXPORT version of IRAF V2.16 supporting PC systems.


  Welcome to IRAF.  To list the available commands, type ? or ??.  To get
  detailed information about a command, type `help <command>'.  To run  a
  command  or  load  a  package,  type  its name.   Type  `bye' to exit a
  package, or `logout' to get out  of the CL.    Type `news' to find  out
  what is new in the version of the system you are using.

  Visit http://iraf.net if you have questions or to report problems.

  The following commands or packages are currently defined:

  (Updated on 2013-12-13)

      adccdrom.   deitab.     images.     mtools.     softools.   upsqiid.
      cfh12k.     esowfi.     kepler.     nfextern.   sqiid.      utilities.
      cirred.     finder.     language.   noao.       stecf.      vo.
      ctio.       fitsutil.   lists.      obsolete.   stsdas.     xdimsum.
      cutoutpkg.  gemini.     mem0.       plot.       system.     xray.
      dataio.     gmisc.      mscdb.      proto.      tables.
      dbms.       guiapps.    mscred.     rvsao.      ucsclris.

ecl>
```

EDIT (this is already done in the VM):

1) login.cl line ~34

set stdimage = **imt4096**

set   imextn = "oif:imh fxf:**fit,fits** ...”

2) include data of Ondrejov observatory in the database, edit file **obsdb.dat** (path: ~miniconda3/envs/iraf38/iraf/noao/lib/obsdb.dat)

```
observatory = "ondrejov"
      name = "Ondrejov observatory"
      longitude = 345:12:59
      latitude = 49:54:38
      altitude = 528
      timezone = -1
```