Problem one

this problem consisted of 5 major parts: generation of a shared Fortran library with a make file, writing a C code that uses said shared Fortran library, and uses it to access the shared Fortran library and generate a executable, se said executable to make data files, and plot them with Gnuplot

To start I had to download the six Fortran routines iridreg, iriflip, irifun, iris, irisub, iritec, and iritest , and associated data files dgfr1945 through dgfr2015.dad in increments of 5, ursi11-ursi22.acs mscat11-mscat22.dat Igrf2020.dat, and igrf2020s.dat, ccir11-ccir22.asc, ig-rz.dat and apf107.dat

I need to install a Linux environment and chose MSY2 in order to install GFortran and make.

I use a make file to compile the Fortran code Lasso, and the legacy subroutines , into a shared library. Then I used gcc to compile the top level C code. Then using Gfortran I linked the object file from the C code with the Fortran library to get an executable code (Note executable must have access to the shared Fortran library). Running the code generated the two data files for each time and date. Then using Gnuplot, I plotted both profiles.

Over the course of this exercise, I became more proficient in C and Fortran. I learned to mate C with Fortran, such as the C equivalent data types to Fortran data types as, as well as dealing with differences in row, column, and numbering. I learned the particular syntax to call Fortran in C, as well as some of complier flags needed for legacy Fortran. I also developed some of the skills necessary to inspect legacy Fortran and determine the required input parameters of a legacy Fortran code.  
I also decided to avoid any alterations to the legacy code to avoid introducing errors. It was a difficult task, but in future after having learned some basics, I expect to be able to have an easer time on a similar problem in the future.