# Clarifications/README

To help users navigate the scripts and make effective use of them, below are some tips and clarifications.

* The scripts assume all the data they will operate on is located in the same directory. To access them from anywhere,
  + move the whole directory with the scripts to somewhere that is in your $PATH
  + in each tcsh script (\*.csh), update setenv SCRIPTS\_HOME pwd to setenv SCRIPTS\_HOME \_\_LOCATION\_OF\_SCRIPTS\_\_
* The scripts have been used on the following operating systems
  + CentOS 6/7, although they should work on most GNU Linux systems
  + Mac OS X 10.10+, assuming it has XCode and most Linux utilities
* The scripts assume you will be running them on an HPC system using
  + PBS batch queuing system
  + all the scripts and necessary software are accessible to the compute nodes via a shared filesystem
* The scripts assume users have the following software
  + OpenBabel
    - only version 2.4.x was tested
  + tcsh shell
    - only version 6.18.x was tested
    - if the location of your ‘tcsh’ is different from /bin/tcsh, you would need to update the scripts with the proper path to ‘tcsh’
  + Python
    - only versions 2.6 and 2.7 were tested
  + Gaussian 09 or 16
    - g09rev[B,D] and g16rev[A-B] were tested
  + Ogolem
    - only the “classic” (x - 2016) version is tested
    - requires Java
  + all packages that interface with Ogolem need to be available. For example,
    - MOPAC to run PM7
    - Orca to run hf3c
    - DFTBplus to run scc-dftb
    - paths to the locations of all these packages need to be specified in the runogolem.csh script
* To calculate the symmetry of molecules, one would likely need to compile the attached ‘symmetry.c’ file by entering ‘gcc -o symmetry symmetry.c’. Since the code was written in 2002, you may get a lot of warning messages if you compile it using recent versions of GNU gcc.
* Rotational constants are calculated using the ‘rot\_conts.py’ script.
  + if your molecule has elements aside from the 10 or so first and second row elements included in the script, you would need to add the atom and its atomic mass in the script.
  + this Python script is slow for processing a large number of files. If users want to compile a much faster C version, they can ask the authors for a more efficient C code.
* The scripts write intermediate data to /tmp assuming it exists and that the user has permission to write to files in that directory. If that is not the case, please change ‘/tmp’ to a a location you have write permissions to.