Skeleton-Module:

* Introduce section: purpose,
* Dependencies, key files
* Functions and subroutines
* Program variables and how to run the program
* testing

GNSS-Qualification:

\*See GNSS-Qualification.txt for running commands

This module is designed to qualify the GNSS acquisition quality in order to access to proceed to apply vertical referencing via GPS-tides or model-tides approach. Uses the GDOP and NSV as parameters for qualification thresholds after NRCAN PPP has been applied.

This module has two main purposes. First is when given a rinex file, it calls the NRCAN PPP script to submit and process the file then download the results locally via HTTP POST method. The second purpose is to qualify the NRCAN PPP output by using the GDOP and NSV parameters and create a sorted list of GNSS readings which pass and fail the threshold test.

This module is dependent on the NRCAN PPP script “csrs\_ppp\_cgi\_browser.py”. This script has been provided by NRCAN. This script requires that WGET and UNZIP be on the Path (Windows) in order to function correctly. The modules require import of os, sys, subprocess, time, argparse, shlex, datetime and csv.

1. apply\_nrcan\_ppp(rnx\_file, output\_path)

* User Parameters:
  + Rnx\_file: full path to the rinex file to apply nrcan PPP
  + Output\_path: full path to download the results of nrcan PPP to
  + Nrcan\_script: name of the nrcan script, default=”csrs\_ppp\_cgi\_browser.py”
  + User\_name: username used by nrcan for tracking. Need to have an account to submit requests to their server, default=[k.a@unb.ca](mailto:k.a@unb.ca)
* Given output path, the script first checks the directory exists then make sure the rinex file is the only item in the directory. This is done to ensure if any errors occur during processing and the script needs to be re-ran, there will be no directory or file related issues.
* The nrcan PPP script is then called using a cmd pipe with the constant preferences prefilled.
* To run this script simply give the full path the the rinex file and full output path
  + Ie. apply\_nrcan\_ppp("C:/PPP/13\_46\_34-2017\_09\_22-gps.17o", "C:/PPP/")
* The nrcan PPP script itself takes in several arguments, most of which have been taken care of for the user. For reference these are the arguments:
  + csrs\_ppp\_cgi\_browser.py --user\_name cgis0172 --lang en --ref ITRF --epoch CURR --mode Static --rnx rnxfile --path work\_dir
  + If any preferences change, update apply\_nrcan\_ppp() function and not the original nrcan PPP function.
  + NOTE: the nrcan PPP script has been documented timing out while processing. This occurs when the wget calls exceed the threshold of default 20 and occurs when there is delays processing due to network errors, server errors or large rinex file inputs. To fix this issue the timeout must be extended by either increasing the wget\_max value or sleepsec values. These are found in the nrcan ppp script under ‘Local Vars’
* Returns 0 if no errors, else returns appropriate error if issues occurred in the apply\_nrcan\_ppp script or nrcan PPP scripts itself.

1. qualify\_ppp\_output(pos\_file\_name, output\_path)

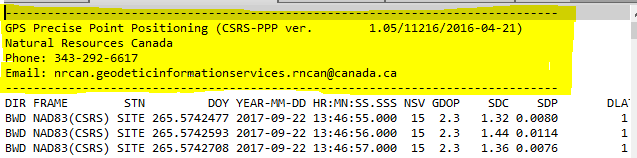
* User Parameters:
  + Pos\_file\_name: filename of the .pos file
  + Output\_path: path of the PPP output where the .pos file resides
* The function first checks if the .pos file exists in the output path. If it exists it opens the .pos for reading, removes the nrcan header (via identify\_pos\_header) which is not needed and separates the contents from the useful information header. A .srt file is created to write the results to
  + identify\_pos\_header(file\_contents)
  + takes in .pos file contents and simply returns the line where the header ends
  + 

Figure 1 .pos file header to be removed

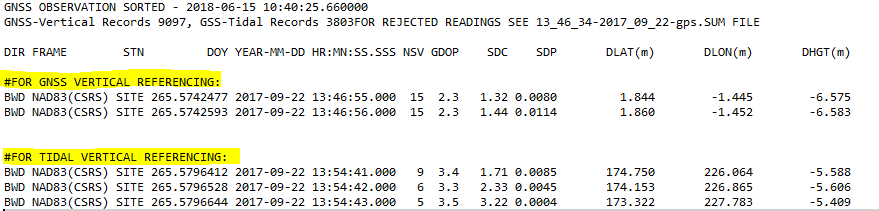
* Function then calls sort\_gnss\_readings:
  + sort\_gps\_readings(header, file\_contents)
  + takes in a header and .pos file contents and sorts the results based on GDOP and NSV values. Returns two lists representing GNSS readings that will need to be vertically referenced using GNSS (threshold met) and another list to be vertically referenced using tides (threshold not met).
  + gdop\_ppp is the GDOP value used for threshold sorting, default=3.5
  + nsv\_ppp is the number of satellites present used for threshold sorting, default=10
* The results of sort\_gps\_readings are then written to the .srt file and saved. Returns 0 if no issues, else returns appropriate error.
* 

Figure 2 .srt file created and results written to

Testing:

Author: Khaleel Arfeen

Email: k.a@unb.ca

Version: 1.0.0