Practical Exercises

EUREF Tutorial: Transformations using PROJ
May 21, 2019, Tallinn

Kristian Evers - kreve@sdfe.dk - @kbevers

Introduction

All the exercises are made as `gie` tests where one or more inputs are missing.

Open the files and replace occurrences of `<your answer here>` with the relevant input. Here's an example:

```
<gie>
1. Set up an operation that returns the input it is given
-------------------------------------------------------------------------------
operation   <your answer here>
tolerance   10 mm

accept      12.3    45.6
expect      12.3    45.6
-------------------------------------------------------------------------------
</gie>
```

Introduction

```
<gie>
1. Set up an operation that returns the input it is given
-------------------------------------------------------------------------------
operation   +proj=latlon    # The latlon "projection" returns the same output as
                            # given in the input
tolerance   10 mm

accept      12.3    45.6
expect      12.3    45.6
-------------------------------------------------------------------------------
</gie>
```

Introduction

Check your answers:

```
> gie exercises\<exercise_name>.gie
--------------------------------------------------------------------------------
Reading file 'exercises\<exercise_name>.gie'
--------------------------------------------------------------------------------
total:  1 tests succeeded,  0 tests skipped,  0 tests failed.
--------------------------------------------------------------------------------
```

Introduction

Your answers may be incorrect:

```
> gie exercises\<exercise_name>.gie
------------------------------------------------------------------------------
Reading file 'exercises\<exercise_name>.gie'
     -----
     FAILURE in <exercise_name>.gie(48):
     expected: 12.3 45.6
     got:      1369229.736757265171   5685937.873489554040
     deviation:  999999999.999000 mm,  expected:  10.000000 mm
------------------------------------------------------------------------------
total:  0 tests succeeded,  0 tests skipped,  1 tests FAILED!
------------------------------------------------------------------------------
```

Exercises

WiFi: KeM-public / PEKPEQYA

- Follow the installation instructions in instructions.md
- Read the “How to fill out PROJ tutorial exercises” section of `instructions.md`
- Do the exercises in this order:
  - `projections1.gie`
  - `projections2.gie`
  - `ellipsoids.gie`
  - `gridshift.gie`
  - `helmert.gie`
  - `conversions.gie`
  - `pipelines.gie`
  - `projections3.gie`
  - `geodesics.gie`

Answers

projections1.gie

1. `+proj=utm +zone=35`
2. `+proj=utm +zone=60 +south`
3. `+proj=utm +zone=35 +ellps=intl`

projections2.gie

1. `+proj=lcc +lat_1=59.8`
2. `+proj=lcc +lat_1=59.8 +k_0=0.99998`
3. `+proj=lcc +lat_1=59.437 +lat_2=60.171`
4. `+proj=lcc +lat_1=59.8 +lat_0=59.8 +lon_0=24.8`
5. `+proj=lcc +lat_1=59.8 +lat_0=59.8 +lon_0=24.8 +x_0=1000000 +y_0=1000000`

ellipsoids.gie

```
1.      +R=1
2.      +ellps=intl
3.      +a=6378388.0 +rf=297.0
4.      +a=6378206.4 +b=6356583.8
```

gridshifts.gie

1. `+proj=hgridshift +grids=BETA2007.gsb`
2. `+proj=vgridshift +grids=egm96_15.gtx`

gridshifts.gie

1. `+proj=hgridshift +grids=BETA2007.gsb`
2. `+proj=vgridshift +grids=egm96_15.gtx`

helmert.gie

```
1. +proj=helmert +x=-97 +y=-103 +z=-120
2. +proj=helmert +x=-81.1 +y=-89.4 +z=-115.8 +rx=0.485
                 +ry=0.024 +rz=0.413 +s=-0.54
                 +convention=position_vector
3a. +proj=helmert +x=582 +y=105 +z=414
                  +rx=-1.04 +ry=-0.35 +rz=3.08 +s=8.3
3b. +proj=helmert +x=582 +y=105 +z=414
                  +rx=1.04 +ry=0.35 +rz=-3.08 +s=8.3
4.  +proj=helmert +convention=position_vector +t_epoch=2010
                  +x=-0.0016  +y=-0.0019  +z=-0.0024
                  +s=2e-05 +dz=0.0001 +ds=-3e-05
```

conversions.gie

1. `+proj=unitconvert +xy_in=m  +xy_out=us-ft`
2. `+proj=axisswap +order=2,1`
3. `+proj=cart +ellps=intl`

pipelines.gie 1

1. ```
   +proj=pipeline +ellps=GRS80
   +step +proj=cart
   +step +proj=helmert +x=100 +y=200 +z=300
         +convention=position_vector
   +step +inv +proj=cart
   ```
   ```

pipelines.gie 2

2. +proj=pipeline
   +step +proj=hgridshift +grids=BETA2007.gsb
   +step +proj=vgridshift +grids=egm96_15.gtx

a: In principle, yes - practically, no.
b: Which grids is applied first depends on how the systems are defined. If a geoid model is fitted to a certain datum, you need to make sure that the input to the vgridshift operation is in that datum. That is your responsibility to check - PROJ can't figure that out on its own. In this case, the EGM96 geoid is fitted to WGS84. The BETA2007 grid results in ETRS89 coordinates which for all but the most high accuracy application can be regarded as equvialent to WGS84. Hence we do the horizontal grid shift first.

pipelines.gie 3

```
+proj=pipeline
     +step +inv +proj=utm +zone=32 +ellps=intl
     +step +proj=cart +ellps=intl
     +step +proj=helmert +x=-81.1 +y=-89.4 +z=-115.8
           +rx=0.485 +ry=0.024 +rz=0.413 +s=-0.54
           +convention=position_vector
     +step +inv +proj=cart +ellps=GRS80
     +step +proj=utm +zone=33
```

pipelines.gie 4

```
    +proj=pipeline
            +step +proj=axisswap +order=2,1
            +step +proj=unitconvert +xy_in=deg +xy_out=rad
            +step +proj=push +v_3 +step +proj=cart +ellps=bessel
            +step +proj=helmert +x=598.1 +y=73.7 +z=418.2
                  +rx=0.202 +ry=0.045 +rz=-2.455 +s=6.7
                     +convention=position_vector
            +step +inv +proj=cart +ellps=GRS80
            +step +proj=pop +v_3
            +step +proj=unitconvert +xy_in=rad +xy_out=deg
            +step +proj=axisswap +order=2,1
```

a: Latitude (deg), longitude (deg), height (m)
b: Latitude (deg), longitude (deg), height (m)
c: Retain the original height of the input coordinate:
Without the push/pop steps the height would be affected by the Helmert step.

projections3.gie

```
1.      +proj=tmerc
2.      +proj=tmerc +k_0=0.9996 +lon_0=27 +x_0=500000
3a.     1 cm
3b.     250 km
3c.     100 km
3d.      10 km
3e.     500  m
3f.       2  m
3g.       5 um
```

geodesics.gie

```
Helsinki,  UTM:  6672241.54   385592.95
Tallinn,   UTM:  6590881.40   372106.37
Mid point, UTM:  6631561.47   378849.66
Mid point, GEO:  59.804039062602   24.840482644156

Meditations:
1.      Probably not
2.      geod +lat_1=60.171 +lon_1=24.938 +lat_2=59.437 +lon_2=24.745 +n_S=2
+ellps=GRS80 -f "%.12f"
3.      Avoids a bit of truncation of the azimuth, and *may* use a slightly
superior algorithm (check the code!)
```