Command-line arguments

- Each program requires certain arguments to define its operation.
- Arguments are explained in the manual pages and in the usage messages.
- Each argument specification begins with a hyphen, followed by a letter, and sometimes a number or character string immediately after the letter.
- Do not space between the hyphen, letter, and number or string.
- Do space between options.
- Example: `pscoast -R0/20/0/20 -G200 -JM6i -W0.25p -B5 -V > map.ps`
Standardized arguments (1/2)

Among the many possible arguments in GMT, 13 have the same meaning for all programs:

- `-B` Defines tickmarks, annotations, and labels for basemaps and axes
- `-H` Specifies that input tables have header record(s)
- `-J` Selects a map projection or one of several non-map projections
- `-P` Selects Portrait plot orientation [Default is landscape]
- `-R` Defines the min. and max. coordinates of the map/plot region
- `-U` Plots a time-stamp, by default in the lower left corner of page
- `-V` Verbose operation
Standardized arguments (2/2)

- \(-X\) Sets the x-coordinate for the plot origin on the page
- \(-Y\) Sets the y-coordinate for the plot origin on the page
- \(-c\) Specifies the number of plot copies
- \(-:\) Input geographic data are (lat,lon) rather than (lon,lat)
- \(-K\) Allows more plot code to be appended to this plot later
- \(-O\) Allows this plot code to be appended to an existing plot
Standardized arguments: exercises

- Verbose:
  pscoast -R0/360/-70/70 -JM6.5i -Ba60f30 -G0 -Dc
  -P -V > GMT_mercator.ps

- Time tag:
  pscoast -R0/360/-70/70 -JM6.5i -Ba60f30 -G0 -Dc
  -P -U > GMT_mercator.ps

- Landscape:
  pscoast -R0/360/-70/70 -JM6.5i -Ba60f30 -G0 -Dc
  -U > GMT_mercator.ps

- Shift by 4 inches vertical:
  pscoast -R0/360/-70/70 -JM6.5i -Ba60f30 -G0 -Dc
  -P -Y4i > GMT_mercator.ps
Overlay and Continue modes

- If a GMT command is **NOT THE LAST** (in a script for instance), then it must contain `-K`, meaning “more postscript code will be appended later”.

- If a GMT command is **NOT THE FIRST**, then it must contain `-O`, meaning “this postscript code is an overlay on top of previous code.”

- Therefore, in a GMT script:
  - The first command contains `-K` only
  - The last command contains `-O` only
  - All commands in between contain `-O` `-K`
Colors in GMT

- GMT allows you to use color and grey scales for drawing lines or filling polygons (landmasses, symbols, etc.)
- For instance:
  - Many GMT commands use \(-G\) to define the fill color (e.g., \texttt{pscoast} for landmasses)
  - Many GMT commands use \(-W\) to define the pen color (and width) for coastlines
    - \texttt{pscoast} uses \(-I\) to define the pen color (and width) for rivers
    - \texttt{pscoast} uses \(-S\) to define the fill color for oceans and lakes.
- Colors are defined with numbers from 0 to 255 that give their intensity
Grey scale in GMT

• White is coded as 0 - Black is coded as 255
• Anything between 0 and 255 is grey
• Example of black landmasses and grey coastlines:
  pscoast -R0/360/-70/70 -JM6.5i -Ba60f30 -Dc -G0 -W4/200 -P > GMT.mercator.ps
Colors in GMT

- Colors in GMT are defined as a mixture of 3 primary colors: Red / Green / Blue (or R/G/B in short).

- The intensity of each of those 3 primary colors ranges from 0 (lowest intensity) to 255 (highest intensity).

- For instance:
  - Red is coded as 255/0/0 (full red, no green, no blue).
  - Yellow is a mixture of red and green, therefore defined as 255/255/0.

- The web link below provides a wide range of RGB color definitions:
  http://stommel.tamu.edu/ baum/GMT/colors/colors.html
Other pen attributes

A pen in GMT has three attributes: width, color, and texture, specified for instance with: 

\[-\text{Wwidth[/color][ttexture][p]}\]

- **Width**: measured in units of the current device resolution. Append \( p \) to specify pen width in points (1/72 of an inch).
- **Color**: see above, gray shade or RGB.
- **Texture**:
  - \( t0 \) = dotted line
  - \( ta \) = dashed line
  - \( t\text{string:offset} = \text{string = length-gap_length-gap, offset from origin} \)

- **Example**: `pscoast -R-140/-50/20/65 -JM6.5i -Ba20f5 -N1/2/255/0/0t20_10.5_10:10 -N2/2/0/to -W1/0/0/255 -P > GMT_mercator.ps`
Plotting text strings

- GMT command is `pstext`

- Input data to `pstext` must contain:
  
  \[ x \ y \ size \ angle \ fontno \ justify \ text \]
  
  - \( x \ y \) = location, in plot units
  - \( size \) = font size in points (e.g. 12)
  - \( angle \) = angle CCW from horizontal
  - \( fontno \) = font number (e.g. 0 for Helvetica)
  - \( justify \) = combination of LCM and TMB:
Plotting text strings: input data

As an external file:

1. Create or edit file with text data, for instance:
   
   ```
   echo "-86 40 14 0 0 CM Purdue" >! mytext.txt
   ```

2. Call `pstext`, for instance after a `pscoast` command:

   ```
   pscoast -R-140/-50/20/65 -JM6.5i -Ba20f5
   -N1/2/255/0/0ta -N2/2/0/to -W1/0/0/255
   -P -K > GMT_mercator.ps
   pstext mytext.txt -R -JM -O >> GMT_mercator.ps
   ```
Plotting text strings: input data
Plotting text strings: input data

Directly:

1. By "piping" text into `pstext`:
   
   ```
   echo "-86 40 14 0 0 CM Purdue" | pstext
   mytext.txt -R -JM -O >> GMT_mercator.ps
   ```

2. Using `<<`:

   ```
   pstext << eof -R -JM -O >> GMT_mercator.ps
   -86 40 14 0 0 CM Purdue
eof
   ```
Exercise

Plot a map of the continental US with the name of each state capital shown at its geographic location. Use file state.capitals, awk, pscoast, and pstext.