Code HW #2:

Submit your solution on Canvas. See end for submission checklist. Make sure to use your own initials in function names.

1) Create a MATLAB or Python function (with your initials!) that calculates the twist for the rotation about a line defined by axis $\boldsymbol{\omega}$ and point on the rotation axis $\mathbf{q}$. The function should return

$$\boldsymbol{\xi} = \begin{bmatrix} -\boldsymbol{\omega} \times \mathbf{q} \\ \boldsymbol{\omega} \end{bmatrix}^{6\times1}$$

for a given rotation axis $\boldsymbol{\omega} \in \mathbb{R}^3$ and point on the rotation axis $\mathbf{q} \in \mathbb{R}^3$. Name this function **`twistr_ckb.m`**. Here's an example of using the MATLAB function:

```
w = [1 2 0]';
w = w/norm(w);
q = [3 -2 1]';
z = twistrCKB(w,q)
z =
  -0.8944
   0.4472
   3.5777
   0.4472
   0.8944
        0
```

2) Create a MATLAB or Python function that returns $\hat{\boldsymbol{\xi}}$ for a given $\boldsymbol{\xi}$. Name this function **`wedge_ckb`** (use your own initials). Here's a MATLAB example of using the function:

```
z_w = wedge_ckb(z)
z_w =
      0         0    0.8944   -0.8944
      0         0   -0.4472    0.4472
-0.8944    0.4472         0    3.5777
      0         0         0         0
```

3) Create a MATLAB/Python function that calculates the matrix exponential of $\hat{\boldsymbol{\xi}}\theta$ for pure rotation. For pure rotation, the matrix exponential of $\hat{\boldsymbol{\xi}}\theta$ can be found using:

$$e^{\hat{\boldsymbol{\xi}}\theta} = \begin{bmatrix} e^{\hat{\boldsymbol{\omega}}\theta} & \left(\mathbf{I} - e^{\hat{\boldsymbol{\omega}}\theta}\right)\hat{\boldsymbol{\omega}}\mathbf{v} \\ 0 & 1 \end{bmatrix}^{4\times4}$$

for a given twist $\boldsymbol{\xi} \in \mathbb{R}^3$ and angle θ. Name your function **`expt_ckb.m`** *(this function should call your* **`expr_ckb`** *and* **`hat_ckb`** *functions)*. Verify your function works correctly by comparing the output from **`expm(wedge_ckb(z)*th`**. Here's an example of using the function:

```
T = expt(z,pi/2)
G = expm(wedge(z)*pi/2)
T =
   0.2000    0.4000    0.8944    2.3056
   0.4000    0.8000   -0.4472   -1.1528
  -0.8944    0.4472    0.0000    4.5777
        0         0         0    1.0000
G =
   0.2000    0.4000    0.8944    2.3056



   0.4000    0.8000   -0.4472   -1.1528
  -0.8944    0.4472    0.0000    4.5777
        0         0         0    1.0000
```

4) The vertices of triangle have initial coordinates $\mathbf{p}_{a1} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T$, $\mathbf{p}_{a2} = \begin{bmatrix} 1 & 0 & 2 \end{bmatrix}^T$, and $\mathbf{p}_{a3} = \begin{bmatrix} 2 & 0 & 1 \end{bmatrix}^T$ with respect to the inertial frame A, assume this triangle rotates about the axis $\boldsymbol{\omega} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T$ which passes through the point $\mathbf{q} = [0 \ \ 1 \ \ 2]^T$. Using your **`twistr_ckb.m`** and **`expt_ckb.m`** sub functions, plot the triangle at rotations from $\theta = 0$ to $3\pi/2$, with step sizes of $\pi/20$. Plot all three sides of the triangle as blue lines. Use an isometric view (in MATLAB this is `view(3)`, for Python's matplotlib use `ax.view_init(elev=30, azim=-37.5))`.

5) A separate triangle is fixed to body frame B, with coordinates $\mathbf{p}_{b1} = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}^T$, $\mathbf{p}_{b2} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T$, and $\mathbf{p}_{b3} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ with respect to body frame B. Assume body frame B rotates about the axis $\boldsymbol{\omega} = [1 \ \ 0 \ \ 0]^T$ which passes through point $\mathbf{q} = [1 \ \ 1 \ \ 1]^T$. The initial configuration of body frame B has a displacement of $\mathbf{p}_{ab} = \begin{bmatrix} 2 & 0 & 2 \end{bmatrix}^T$ and a rotation of 90 degrees about the z-axis with respect to inertial frame A. Using MATLAB/Python, find $\mathbf{g}_{ab}(\theta)$ and plot the triangle at rotations from $\theta = 0$ to $3\pi/2$, with step sizes of $\pi/20$. Plot all 3 sides of the triangle as red lines on the same figure as 4) above.

6) Your main MATLAB/Python script should output the figure from 4) and 5) as well as $\mathbf{g}_{ab}(0)$ from 5). Your main scripts file should only create the figure and output a 4x4 matrix (nothing else should be outputted to the command window).

HW #2: WHAT TO SUBMIT

Submit ***individual files*** on Canvas (not zipped) using the naming convention given. Check Canvas for the due date and time.

A. **HW02_Lastname_plot.pdf**. A single PDF or Word document of the assigned plot *(use your last name, not mine!)*. Use "Copy Figure" from the "Edit" figure menu and paste it into a Word document or similar. Do not include figure windows, menus, etc. Avoid shrinking the plot when inserting it into Word (or similar) and set the font size to 12 pt or higher.
B. **`HW02_Lastname_main.m/.py:`** your main script
C. **`twistr_CKB.m /.py:`** your twist function file
D. **`wedge_CKB.m/.py:`** your wedge function file
E. **`expt_CKB.m/.py:`** your exponentiated twist function file
F. **`expr_CKB.m/.py:`** your exponentiated rotation function file
G. **`hat_CKB.m/.py:`** your hat function file