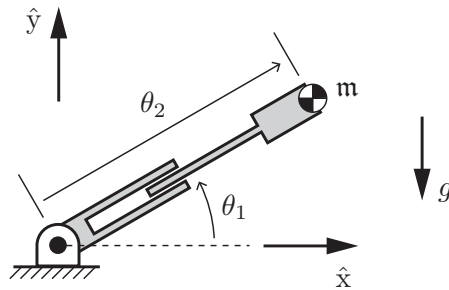1. Use Lagrange's formulation to derive the dynamics for the planar RP robot shown below, which has a single point mass $\mathfrak{m}$ at the tip of the robot. Show your derivation. Write the resulting equations in the form $\tau = M(\theta)\ddot{\theta} + c(\theta, \dot{\theta}) + g(\theta)$, including the 2×2 mass matrix. Indicate any Coriolis terms and any centripetal terms in the velocity-product term $c(\theta, \dot{\theta})$.

2. Consider the equation $Ax = b$, where $A \in \mathbb{R}^{2\times 3}$, x is a 3-vector, and b is a 2-vector. (If you'd like, you could consider the analogy where A is a Jacobian, x is a set of joint velocities, and b is a twist.) Let

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix},$$

and $b = (7, 8)$.

   (a) Solve for x that satisfies $Ax = b$ while minimizing $\|x\|$. Feel free to use any software you'd like. Call this solution x^*.

   (b) Mathematically express the space of solutions x to $Ax = b$ in the form $x^* + cv$, where v is a 3-vector and c is a scalar. The vector v should be expressed numerically.

3. Starting from the function `IKinBody`, write a new function, `IKinBodyIterates`. This function prints out a report for each iteration of the Newton-Raphson process, for iterates 0 (the initial guess) to the final solution. Each iteration reports the iteration number i, the joint vector θ^i, the end-effector configuration $T_{sb}(\theta^i)$, the error twist $\mathcal{V}_b$, and the angular and linear error magnitudes, $\|\omega_b\|$ and $\|v_b\|$ (something like the table at the end of Chapter 6.2.2). For a four-joint robot, a typical iterate might look like:

```
Iteration 3:

  joint vector:
  0.221, 0.375, 2.233, 1.414

  SE(3) end-effector config:
   1.000  0.000  0.000  3.275
```

```
0.000  1.000  0.000   4.162
0.000  0.000  1.000  -5.732
0      0      0       1


error twist Vb:
0.232, 0.171, 0.211, 0.345, 1.367, -0.222


angular error magnitude ||omega_b||:  0.357


linear error magnitude ||v_b||:  1.427
```

Test your new function for the UR5 robot of Chapter 4.1.2 (Figure 4.6) and a goal configuration of

$$T_{sd} = \begin{bmatrix} 0 & 1 & 0 & -0.5 \\ 0 & 0 & -1 & 0.1 \\ -1 & 0 & 0 & 0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where linear distances are in meters. Use $\epsilon_\omega = 0.001$ rad ($0.057°$) and $\epsilon_v = 0.0001$ (0.1 mm). Choose an initial guess θ^0 so that the numerical inverse kinematics converges in 3–5 iterations. Use the V-REP Scene 1 to visualize the robot at its initial configuration and the final configuration returned by the IKinBodyIterates function. Turn in the two images of the robot from V-REP (e.g., by screen capture) and the printed output log with all iterates.

Keep in mind that if your initial guess is far from the solution, the iterates may thrash about a bit before settling to a solution. You could force any joint angles outside a certain range (e.g., 0 to 2π) to be mapped back into that range.