1) What problem can occur when using fixed angles? How can it be solved?
   a) Gimbal lock:
   \[ \begin{array}{c}
   \cos(\theta/2) = 0.966 \\
   \sin(\theta/2) = 0.126 \end{array} \]
   \[ \begin{array}{c}
   v = \frac{1}{\sqrt{(3,1,1)^2 + (3,-2,1)^2)}} \cdot \sin(\theta/2) \\
   = (0.128, -0.126, 1.012) \]

2) What is the quaternion that represents a rotation of 30 degrees about axis (3, -2, 1)?
   Rotation as quaternion:
   \[ E(\cos(\theta/2), \sin(\theta/2))(x,y,z) \]
   \[ \begin{array}{c}
   x = \cos(15) = 0.966 \\
   v = \frac{1}{\sqrt{(3,1,1)^2 + (3,-2,1)^2)}} \cdot \sin(15) \\
   = (0.184, -0.126, 0.126) \]

3) The interpolation of two quaternions \( q_a \) and \( q_b \) is given by:
   \[ q(t) = (q_a \sin((1-t)\theta)) + q_b \sin(t\theta) \]
   where \( \theta \) is the angle between \( q_a \) and \( q_b \).
   What is the quaternion \( q(t) \) that represents the rotation \( \theta \) of the way between \( (0.924, 0.171, 0.347, 2.0) \)
   and \( (0.866, 0.0, 0.5, 0.0) \)?

   Compute \( q(0.5) \) via the inner product:
   \[ \cos(\theta) = \langle (0.924, 0.171, 0.347, 2.0), (0.866, 0.0, 0.5, 0.0) \rangle \]
   \[ = 0.971 \]
   \[ q(0.5) = \langle (0.924, 0.171, 0.347, 2.0) \rangle \sin(0.6 \cdot 15.787) \]

\[ \frac{1}{\sqrt{(0.924, 0.171, 0.347, 2.0)^2 + (0.866, 0.0, 0.5, 0.0)^2}} \sin(0.4 \cdot 15) \]
4) For a given curve path \( p(s) \), how can we compute the
view direction and up vector?

- Normal frame:
  - View direction: \( p'(s) \)
  - Up vector: \( p'(s) \times p''(s) \)

5) When does the frame-frame-based calculation of
the up vector fail?

1) When curvature is zero:

2) When second derivative switches direction:

6) What is the difference between global
 transformations and FFD?

Global transformations apply a matrix (or a
series of matrices) to an object to deform it. The matrix can be based on the coefficients of the curve it is applied to. FFD defers
definition by using Bezier-based deformable
functionals.

7) What is the difference between inverse and
forward kinematics?

In forward kinematics, joint parameters are
specified explicitly, whereas in inverse kinematics
the location of an end effector is specified and
the system computes joint parameters.
8) What methods can help speed up collision detection:
- bounding volumes
  - bounding sphere
  - bounding boxes
  - bounding slab
  - convex hull
- vector inside object test

9) Assume a particle is moving towards a horizontal plane with a velocity of 
\[ \mathbf{v} = \begin{pmatrix} \frac{\pi}{2} \\ \frac{\pi}{2} \end{pmatrix} \]. Assuming there is no damping, what is the direction of the particle after the collision?

\[ \mathbf{v} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \]

\[ \mathbf{v}_c = \mathbf{v} - k \cdot \mathbf{n} = \left( \begin{pmatrix} \frac{\pi}{2} \\ \frac{\pi}{2} \end{pmatrix} - k \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) \]

\[ = \begin{pmatrix} \frac{\pi}{2} \\ \frac{\pi}{2} - k \end{pmatrix} \]

\[ = \begin{pmatrix} \frac{\pi}{2} \\ \frac{\pi}{2} - 1 \end{pmatrix} \]