CMSC 425: First Midterm Exam

This exam is closed-book and closed-notes. You may use one sheet of notes (front and back). Write all answers in the exam booklet. You may use any algorithms or results given in class. If you have a question, either raise your hand or come to the front of class. Total point value is 100 points. Good luck!

Problem 1. (20 points, 2-6 points each) Short answer questions. Explanations are not required but may be given for partial credit.

1.1 What does it mean to declare a Unity game object to be "Kinematic"?

1.2 Let a and b be points, and let $\vec{v} = b - a$ be a vector. Express $a - 3\vec{v}$ as an *affine combination* of a and b. Is this a *convex combination*? Explain briefly.

1.3 What is the difference between the following two Unity commands with respect to how they act on a game object?

```
transform.Rotate(Vector3.up * Time.deltaTime, Space.Self);
transform.Rotate(Vector3.up * Time.deltaTime, Space.World);
```

1.4 You are given a 4×4 grid indexed by rows i and columns j. You want to enumerate the cells of this grid, from 0 to 15, in three orders: *Row-major*, *Hilbert*, and *Morton* (or Z). In the figure below, we have numbered the first eight entries for you. In each case, complete the ordering by filling in the remaining numbers (8 through 15).

Row major

$i = 0$	0	1	2	3
1	4	5	6	7
2				
3				
$j =$	0	1	2	3

Hilbert

$i = 0$	0	3	4	5
1	1	2	7	6
2				
3				
$j =$	0	1	2	3

Morton (Z)

$i = 0$	0	1	4	5
1	2	3	6	7
2				
3				
$j =$	0	1	2	3

Figure 1: Problem 1.4: Ordering cells of a grid.

1.5 Without *blended skinning*, what might go wrong with the mesh if when a joint is rotated?

Problem 2. (20 points) You have been assigned the job of creating a new type of collider for your company's game engine, a *hemisphere*. The hemisphere is defined by a center point $c = (c_x, c_y, c_z)$, a scalar radius $r > 0$, and a unit vector $\vec{u} = (u_x, u_y, u_z)$ that points in the direction of the empty portion of the hemisphere (see the figure below(a)).

2.1 (10 points) Present a test (using the operations of affine and Euclidean geometry) that determines whether a point $p = (p_x, p_y, p_z)$ lies within the hemisphere collider defined by c, $\vec{u}$, and r. **Hint:** If you cannot see how to solve this. I will give half credit if you solve the special case where $\vec{u} = (0, 1, 0)$, that is, the southern hemisphere (see the figure (b)) assuming Unity's coordinate convention.

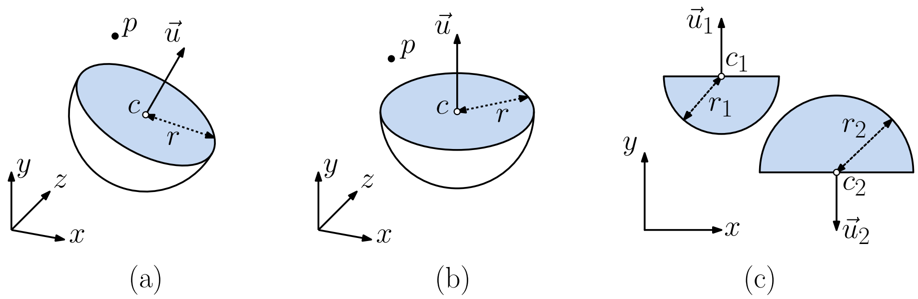

Figure 2: Problem 2: Hemisphere collider.

2.2 (5 points) Suppose that you are given two hemisphere colliders $(c_1, \vec{u}_1, r_1)$ and $(c_2, \vec{u}_2, r_2)$, where $\vec{u}_1 = (0, 1, 0)$ and $\vec{u}_2 = (0, -1, 0)$ (see the figure below). In other words, collider-1 has only its southern hemisphere and collider-2 has only its northern hemisphere.

True or false: If *both* of the following conditions are *satisfied*, the colliders *intersect*.

Condition 1: $c_{1y} \geq c_{2y}$

Condition 2: $\mathrm{dist}(c_1, c_2) \leq r_1 + r_2$

2.3 (5 points) Same set-up as 2.2. **True or false**: If *any* of the conditions *fail*, the colliders *do not* intersect.

In both 2.2 and 2.3, **explain your answer**. You may assume that the degenerate case of colliders touching without penetrating does not occur.

Problem 3. (20 points) Let us consider yet another variant of the projectile-shooting problem. A projectile is launched from a location that is h meters above the ground. Let us assume Unity's convention, where the ground is the (x, z)-plane. Let (p_x, h, p_z) denote the point where the projectile is launched. Let $\vec{v}_0 = (v_{0,x}, v_{0,y}, v_{0,z})$ be the projectile's initial velocity (see the figure below). Someone has placed a circular portal of radius r centered at point $c = (0, c_y, c_z)$ on the (y, z)-coordinate plane.

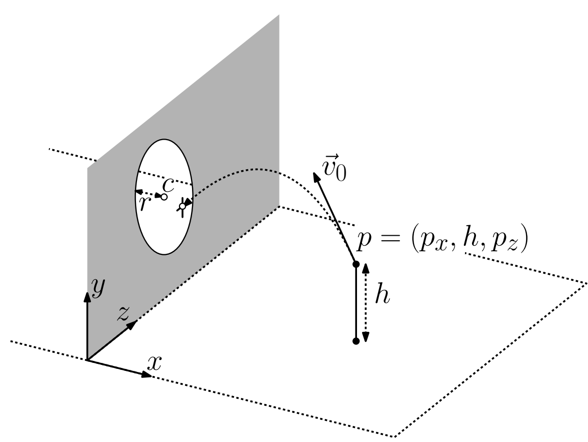

Figure 3: Problem 3: Projectile shooting.

In this problem we will determine whether the projectile passes through the portal. As in the lecture, you may assume that $g \approx 9.8\text{m}/\text{sec}^2$ denotes the acceleration due to gravity. **Note:** You may assume that p lies in the first quadrant (that is, p_x, h, and p_z are all positive).

3.1 (5 points) Give a mathematical expression to determine the time t^* when the projectile hits the (y, z)-coordinate plane. (If there are conditions under which a solution does not exist, explain what they are and how to compute them.)

3.2 (10 points) If a solution to part 3.1 exists, give a mathematical expression for the coordinates of the point q where the projectile hits this plane.

3.3 (5 points) Given the answers to 3.1 and 3.2, determine whether the projectile flies through the portal.

Problem 4. (25 points) Your new game features a quadcopter. Let's consider the problem in a 2-dimensional setting. The copter's reference pose is shown in the figure below left. Its center is at $b = (10, 7)$, and the propellers are aligned as shown. (We are only interested in the top propeller, labeled c.) There are four arms leading out to the propellers, each of length 8. Each propeller blade is of length 5. Let p be a point at the end of the top propeller. Our objective is to derive an affine transformation that maps p to its new position when the copter moves and rotates. There are three frames: the world frame a, the copter frame b, and the top-propeller's frame c. (Note that x points up and y points to the left in this last frame.)

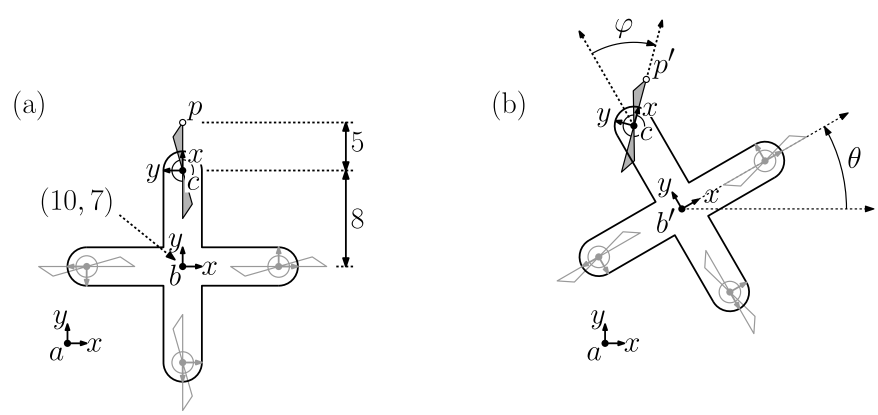

Figure 4: Problem 4: Quadcopter simulator.

4.1 (3 points) Express the position of p (in the reference pose) with respect to each of the coordinate frames as a 3-element vector in *homogeneous coordinates*:

4.2 (10 points) Express the following local-pose transformations, each as a homogeneous a 3×3 matrix. Assuming the reference pose in (a) of the figure. (If you cannot determine the actual matrix, you can express your answer as the product of matrices for partial credit.)

(i) $T_{[b \leftarrow c]}$ (propeller coordinates to copter coordinates)

(ii) $T_{[a \leftarrow b]}$ (copter coordinates to world coordinates)

4.3 (2 points) Using the above matrices, explain how to compute the matrices $T_{[a\leftarrow c]}$ and $T_{[c\leftarrow a]}$. (**Hint**: Use matrix multiplication and matrix inversion.)

4.4 (10 points) Consider the motion shown in the earlier figure (b):

- the propeller rotates *clockwise* by an angle of φ (radians) about its center c
- the copter rotates *counterclockwise* by an angle of θ (radians) about its center b
- the copter translates to a new center point $b' = (b'_x, b'_y)$.

Present a transformation that maps $p[a]$ in the reference pose to its new position $p'[a]$ as a result of this motion. In both instances p is expressed relative to the world's coordinate frame (a).

Note: Present your answer as a sequence of matrices to multiply together. Whenever possible, refer to matrices by *name* (e.g., $\mathrm{Rot}(\theta)$ for rotation by θ, or $\mathrm{Tr}(\vec{v})$ for a translation by offset $\vec{v}$) and *not* as a 3×3 matrix.

Problem 5. (15 points) In the famous game *Portal*, the player can create two virtual holes on a wall, a blue *entry portal* and an orange *exit portal*. Any point q that moves into the entry portal appears instantaneously at the corresponding point q' at the exit portal (see the figure below (a)).

This concept can be implemented using affine geometry. Let's see how.

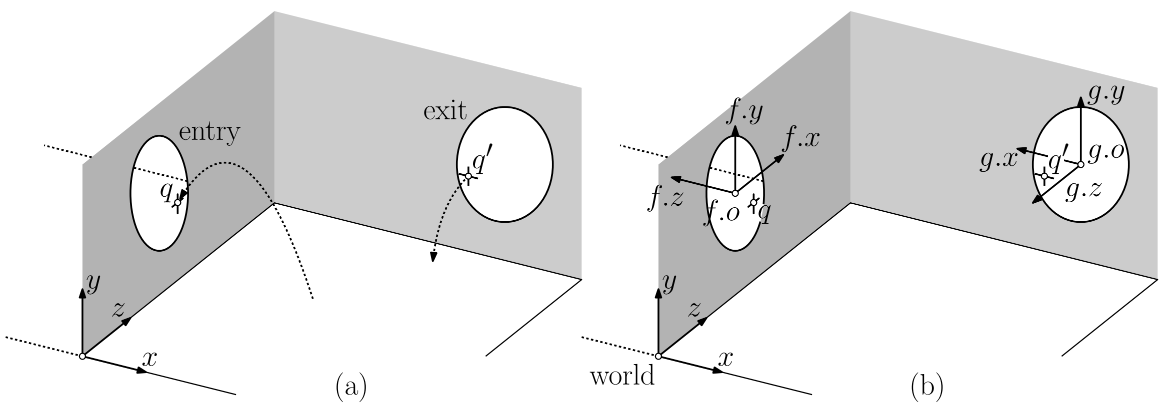

Figure 5: Problem 3: Basic Portal.

Let's consider the setup shown in the figure. We create a frame f at the entry portal, where the z-axis points out from the room (into the portal), and we create a frame g at the exit portal, where the z-axis points into the room (outwards from the portal) See the figure above (b). Let $f.o$ be the center of the portal, and let $f.x$, $f.y$, and $f.z$ be the basis vectors of the entry frame. Define $g.o$, $g.x$, $g.y$, and $g.z$ similarly for the exit frame. Assume that these are all expressed in *homogeneous coordinates* with respect to *world frame*.

5.1 (5 points) Let $T_{[w\leftarrow f]}$ be the transformation that converts a point in f's coordinate frame to the world's coordinate frame. Explain how to compute this transformation as a 4×4 matrix. (In particular, what are its columns and in what order do they appear?)

5.2 (3 points) Do the same for $T_{[w\leftarrow g]}$ the transformation that converts a point in g's coordinate frame to the world's coordinate frame.

5.3 (2 points) Given your answers to 5.1 and 5.2, explain how to obtain the reverse transformations $T_{[f \leftarrow w]}$ and $T_{[g \leftarrow w]}$. (You can explain in terms of matrix inversion.)

5.4 (5 points) Given the matrices you have derived in parts 5.1–5.3, derive a transformation that maps a point q on the entry portal to the corresponding point q' on the exit portal. Assume that both points are represented with respect to the world coordinates. That is, derive a matrix M such that $q'_{[w]} = M q_{[w]}$. (Explain your answer.)

(**Hint**: You can answer each part independently. The answers are all very short.)