# CMSC 425: Lecture MR
# Midterm Review

**Overview:** So far this, we have provided an introduction to many of the principles that underly computer game programming. Our guiding principle has been to understand the computational techniques and methods that are fundamental to programming games and game engines. Here is a list of topics that will be covered on the midterm exam. (If anything seems to be missing, please check with me.)

- *Computer Game and Graphics System Architectures*: The basic software structure of a typical game engine, the design of modern graphics processing units (GPUs), the graphics pipeline, and the impact of this design on game engine software.
- *Introduction to Unity*: The major elements of the Unity game engine, including scenes, packages, prefabs, game objects, components, and scripts. We also discussed other topics such a rigid bodies, classes (Vector3, Ray, Quaternion), initialization and update functions, object and component access, colliders and triggers, kinematic and static, instantiating prefabs, and coroutines.
- *Basics of Linear and Affine Geometry:* Affine geometry (points, vectors, affine and convex combinations), Euclidean geometry (dot and cross product, orientation), three-dimensional rotation (Euler angles and quaternions), homogeneous coordinates, affine transformations, coordinate frames and change of coordinate transformations.
- *Rotation:* Euler angles and quaternions and their pros and cons. Modeling rotations through quaternions and (spherical) interpolation.
- *Enclosures and Spatial Data Structures*: Enclosures for collision detection (e.g., AABBs, spheres, capsules, k-DOPs). Spatial data structures such as grids, quadtrees, and kd-trees. Methods for storing grids and linearization (using row-major, Hilbert, and the Morton orders).
- *Animation and Skinning*: Skeletal (tree) structure as joints and bones, bind (reference) pose, local-pose transformations and their inverse, change of coordinates, forward kinematics, linear-blend skinning.