Geometric Modeling

(Basic Geometric Operations & Intersection Calculations)

1. Introduction to Geometric Modeling

Geometric modeling is one of the most fundamental topics in computer graphics. It refers to the process of representing real-world objects using mathematical descriptions so that a computer can understand, store, and display them. In the real world, humans perceive objects as complete shapes, but a computer cannot directly understand shapes. Instead, it understands numbers, coordinates, and equations. Therefore, every object that appears on the screen is first converted into a mathematical form. This mathematical representation defines the shape, size, position, and orientation of an object in a graphical scene. Geometric modeling forms the base of almost all graphics operations, including rendering, animation, collision detection, and ray tracing. In examinations, students are often asked to define geometric modeling and explain why it is important in computer graphics systems.

2. Basic Geometric Primitives

Geometric primitives are the simplest elements used to construct complex graphical objects. No matter how detailed or realistic a model looks, it is ultimately built from a combination of these basic primitives. Understanding geometric primitives is essential because they serve as the foundation of geometric modeling.

2.1 Point

A point represents a single location in space and does not have any size, shape, or area. It only defines position. In a two-dimensional space, a point is represented by two coordinates, usually written as (x, y). In a three-dimensional space, a point is represented by three coordinates, written as (x, y, z). Points are commonly used to represent vertices of polygons or positions where events occur, such as a mouse click on the screen. In exams, students may be asked to explain how points are represented in 2D and 3D space.

2.2 Line and Line Segment

A line is a straight path that extends infinitely in both directions, while a line segment is a finite part of a line with two endpoints. In computer graphics, line segments are used instead of infinite lines because graphical objects have finite boundaries. Line segments are used to form the edges of polygons and wireframe models. Understanding the difference between a line and a line segment is important from both conceptual and examination perspectives.

2.3 Plane

A plane is a flat surface that extends infinitely in all directions. It is commonly used to represent surfaces such as floors, walls, and tabletops in graphical scenes. Planes play a very important role in geometric operations, especially in intersection calculations, where rays or lines are tested to see where they intersect a surface. Questions related to planes often appear in exams in the context of intersection or visibility.

2.4 Polygon

A polygon is a closed shape formed by connecting multiple line segments. Polygons are one of the most important primitives in computer graphics because most objects are ultimately represented using polygons. Among all polygons, the triangle is the most widely used because it is always planar and can easily represent complex surfaces. Modern graphics hardware, such as GPUs, is highly optimized for processing triangles. In exams, students are often asked why triangles are preferred in computer graphics.

3. Coordinate Systems

Coordinate systems are used to define the **position of objects** in space.

3.1 2D Coordinate System

Coordinate systems are used to define the position of objects in space. Without a coordinate system, it would be impossible to describe where an object is located. In a two-dimensional coordinate system, positions are defined using the x-axis and y-axis. This type of coordinate system is commonly used in simple graphics applications such as user interfaces and 2D drawings.

3.2 3D Coordinate System

- x-axis
- y-axis
- z-axis (depth)

In a three-dimensional coordinate system, an additional axis called the z-axis is introduced. The z-axis represents depth and indicates how far an object is from the camera or viewer. Three-dimensional coordinate systems are essential for games, animations, simulations, and CAD applications. In examinations, students are frequently asked to differentiate between 2D and 3D coordinate systems and explain the role of the z-axis.

4. Mathematical Representation of Geometric Objects

Geometric objects are represented using **mathematical equations**.

4.1 Point Representation

A point is represented directly by its coordinates in space.

P = (x, y) in 2D

P = (x, y, z) in 3D

4.2 Line Representation (Parametric Form)

A line is often represented using a parametric equation, which defines all the points that lie on the line.

$P(t) = P_0 + tD$

Where: - P_0 = starting point - D = direction vector - t = parameter

Use: - Rays - Edges - Intersection calculations

4.3 Plane Equation

A plane is represented using a linear equation of the form:

$Ax + By + Cz + D = 0$

This equation describes all the points that lie on the plane. Plane equations are widely used in graphics applications such as line–plane intersection, visibility testing, and cutting operations. In exams, students are often expected to explain the purpose of plane equations rather than derive them mathematically.

5. Intersection Calculations

Intersection calculation refers to the process of determining whether two or more geometric objects intersect and, if so, where the intersection occurs. Intersection calculations are extremely important in computer graphics because they help determine visibility, detect collisions, and support realistic rendering techniques. Many core graphics algorithms depend on intersection testing.

5.1 Line–Line Intersection

Line–line intersection determines whether two lines intersect, are parallel, or overlap. This type of intersection is commonly used in wireframe rendering and edge collision detection. Although the mathematical details can be complex, students are usually required to understand the concept rather than perform detailed calculations in exams.

5.2 Line–Plane Intersection

Line–plane intersection determines the point at which a line intersects a plane. This concept is very important in computer graphics. For example, when a ray is cast from a camera and hits a surface, a line–plane intersection calculation is performed. This technique is widely used in ray casting, visibility testing, and object picking using a mouse. Examination questions often ask students to explain line–plane intersection along with its applications.

5.3 Ray–Object Intersection

Ray–object intersection is one of the most important concepts in modern computer graphics. A ray starts from a point and travels infinitely in a specific direction. Ray–object intersection determines which object a ray hits first. This concept forms the foundation of ray tracing, where rays are used to simulate light behavior. It is also used in shadow computation, reflection, refraction, and visibility determination. From an exam point of view, students should remember that ray–object intersection is the base of ray tracing.

6. Inside–Outside Test

The inside–outside test is used to determine whether a given point lies inside or outside a polygon. This test is important in many graphics applications, such as object selection, collision detection, and polygon filling. For example, when a user clicks on the screen, an inside–outside test is used to determine whether the click occurred inside a particular object. In exams, this topic usually appears as a short conceptual question.

7. Applications of Geometric Modeling

Geometric modeling is widely used in many real-world applications. In games, it is used for collision detection and object interaction. In CAD systems, it helps designers create accurate models of buildings and machines. In animation and visual effects, geometric modeling is used to construct characters and environments. Ray tracing techniques rely heavily on geometric modeling to simulate realistic lighting. Virtual reality systems also depend on accurate geometric models to create immersive experiences. In exams, students are often asked to explain the applications of geometric modeling in computer graphics.