Mathematics for Computer Graphics

Contents

Preface

Books

- Foundations of Game Engine (Eric Lengyel)

- Physically Based Rendering (Matt Pharr, Wenzel Jacob, and Greg Humphreys)

- Computer Graphics Principles and Practice (John F. Hughes, Andries Van Dam, Morgan Mcguire, David F. Sklar, James D. Foley, Steven K. Feiner, and Kurt Akeley)

- Vector Calculus, Linear Algebra, and Differential Forms (Hubbard)

- Linear Algebra (Gilbert Strang)

- Calculus (Spivak)

- A Tour of C++ (Bjarne Stroustrup)

Environment

- IDE: Visual Studio Code

- Language: C++ (MinGW-w64, cplusplus, and cppreference)

- Version Control: Git

Chapter 1

Vectors

N-dimensional Vector

$$V = (V_0, V_1, V_2, \ldots V_{n-1})$$

3D Vectors

In Cartesian Coordinates, the numbers making up a 3-dimensional vector are called the x, y, and z vector's components because they correspond to distances measured parallel to the x, y, and z axes.

$$V = (V_x, V_y, V_z)$$

Magnitude

Size of the vector.

Magnitude of an ND vector

$$V = \sqrt[2]{\sum_{i=0}^{n-1} V_i^2}$$

Magnitude of a 3D vector

$$V = \sqrt[2]{V_x^2 + V_y^2 + V_z^2}$$

Unit Vector

A unit vector is a vector with magnitude and direction divided by its own magnitude, yielding the direction and a magnitude of value 1. The process of turning a vector into a unit vector is called normalization, the vector is called a normalized vector.

$$V_u = \frac{V}{\|V\|}$$

Multiplication

It changes the Magnitude while keeping the Direction parallel to the original vector. When the scalar is negative, the new vector points in the opposite direction.

$$V(s) = (V_x(s) + V_y(s) + V_z(s) \ldots + V_{n-1}(s))$$

Addition

$$V1 + V2 = (V1_x + V2_x, V1_y + V2_y, V1_y + V2_y \ldots V1_{n-1} + V2_{n-1})$$