1. Record Nr. UNINA9910364950403321

| | |
|---|---|
| Autore | Sung Kelvin |
| Titolo | Basic Math for Game Development with Unity 3D : A Beginner's Guide to Mathematical Foundations / / by Kelvin Sung, Gregory Smith |
| Pubbl/distr/stampa | Berkeley, CA : , : Apress : , : Imprint : Apress, , 2019 |
| ISBN | 9781484254431<br>1484254430 |
| Edizione | [1st ed. 2019.] |
| Descrizione fisica | 1 online resource (414 pages) |
| Disciplina | 794.815 |
| Soggetti | Computer games—Programming<br>Computer science—Mathematics<br>Game Development<br>Math Applications in Computer Science |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Nota di bibliografia | Includes bibliographical references. |
| Nota di contenuto | Chapter 1: Introduction and Learning Environment -- Chapter 2: Intervals and Bounding Boxes -- Chapter 3: Distances and Bounding Spheres -- Chapter 4: Vectors -- Chapter 5: Vector Dot Products -- Chapter 6: Vector Cross Products and 2D Planes -- Chapter 7: Conclusion. |
| Sommario/riassunto | Use Unity-based examples to understand fundamental mathematical concepts and see how they are applied when building modern video game functionality. You will gain the theoretical foundation you need, and you will know how to examine and modify an implementation. This book covers points in a 3D Cartesian coordinate system, and then discusses vectors and the details of dot and cross products. Basic mathematical foundations are illustrated through Unity-based example implementations. Also provided are examples showing how the concepts are applied when implementing video game functionality, such as collision support, motion simulations, autonomous behaviors, shadow approximations, and reflection off arbitrary walls. Throughout this book, you learn and examine the concepts and their applications in a game engine. You will: Understand the basic concepts of points and vectors and their applications in game development Apply |

mathematical concepts to modern video game functionality, such as spherical and box colliders Implement autonomous behaviors, including following way points, facing a target, chasing an object, etc.