I got you fam!

1. Beginning C++ Through Game Programming (2014) - After you finish this, build your own text adventure game.
2. Thinking Like a Programmer (2012) - Learn to solve problems by breaking them down like a programmer.
3. Exercises for Programmers (2015) - Work through Chapters 1 - 7 by applying what you've learned from *Thinking Like a Programmer*.
4. The Black Art of Multiplatform Game Programming (2014) - Clone every game from this list.
5. C++ Primer (2012) - C++ Deep dive with C++11
6. 3D Math Primer for Graphics and Game Development (2011) - Math prep / visit Khan Academy for help if you need it.
7. Game Coding Complete (2012) - Recommended by others in comments.
8. Game Programming Algorithms and Techniques (2013) - Build a game applying this stuff.
9. Game Programming Patterns (2014) - Build another game applying this stuff.
10. Game Engine Architecture (2014) - Attempt to build your own engine with this stuff, but it's really more about understanding how game engines are built.

**Q: What to do after finishing a book?**

**A:** Apply what you've learned! Don't just finish a book and then start next one. Make a little game with what you've learned from that book.

Also these lists:

1. A Study Path for Game Programmer - Don't use this as a sequential guide, but as a reference. For example, if you want to learn more about AI jump to the AI section and check out the recommended books.
2. Blizzard Software Engineering Reading (Associate) - Again don't use this as a step by step guide, but use it as: "This is what I should know at this level."
3. Blizzard Software Engineering Reading (Mid-level) - "This is what I should know at this other level."
4. Blizzard Software Engineering Reading (Senior) - "This is what I should know at this other other level."
5. Game Developer Roadmap - What you need to know to be a game developer (very detailed and overwhelming!).

Books below are not published yet, but I do plan on checking them out.

1. Game Programming in C++: Creating 3D Games (2018) - From the Author of *Game Programming Algorithms and Techniques (2013)*
2. Fundamentals of C/C++ Game Programming: Using Target-based Development on SBCs (2018)

**Update 1:** I've received a few PMs why I didn't suggest books on specific game engines like Unity or Unreal Engine. And reason why is pretty simple... Having a fundamental understanding how these tools (game engines) are working under the hood will yield better results in the long run. Game Engines, much like web frameworks, will change. Right now everyone's using Unity and Unreal, but tomorrow it might be Amazon Lumberyard or Godot. And yeah, these engines work pretty much the same. I do agree with you, but understanding those fundamentals is crucial.

**Update 2:** For those of you who wish to get some sort CS background check out Teach Yourself Computer Science.

**Update 3:** Learning on your own through books can be tough. I wanted to add some quick reading for that provide great advice on learning programming and learning in general.

**General Learning Advice:**

- How to Study

- Learning By Doing
- How to Read a Book
- 10 Rules for Good (and Bad) Studying

**Programming Learning Advice:**

- Learning Techniques for Programmers, by Programmers
- Techniques for Efficiently Learning Programming Languages
- Tips for Self-Learning Programming