I'm not a robot

C is a general-purpose, procedural computer programming language invented in 1972 (a looong time ago). It is considered the base for other modern programming languages and is also known as the “mother language.” C can be considered the “parent” of other languages like C++. C was initially developed by Dennis Ritchie as a system programming language to write operating systems. C is a very flexible and versatile programming language that allows maximum control with minimal commands. There are great C programming books for beginners. Choosing the right book is important as it can guide your learning.In this article, we have compiled a list of best C books for beginners through a collection of book reviews. Each book review will highlight the taste of the book, the contents covered, and how it can benefit you.Why learn C?Is C a good programming language to start with? It’s an understandable question. C is still a commonly-taught programming language, particularly for its historical value. The conventions in C are common in other languages.Here are my reasons why learning C is a great investment in your future: Influence: C has been around for 30 years and has influenced many languages, not limited to obvious C spin-offs like C# or C++.Beginner Friendly: It’s the basic language of a lot of advanced computer languages. Once you master C programming, you can easily learn another specified language.Mid-Level: C is a middle-level language and is reasonably close to the machine. Learning C will teach you the underlying principles of computer architecture. You tend to get a better picture of advanced topics like exactly how networking works. Speed: C is wicked fast in terms of execution time. Programs written and compiled in C execute much faster as compared to any other programming language.Great for Hardware: C is extensively used in Embedded Programming or micro-controller programming where C program is used to control micro-controllers. This type of programming is widely used in auto-motives, Robotics, Hardware, etc.Procedural: Learning C will also teach you to write clean procedural code in an imperative paradigm.Significant: The C programming language is recognized worldwide and used in a number of applications, including advanced scientific systems and operating systems. To help you in selecting a well-structured and latest C book for beginners, we have narrowed it down to the top 10 best C books for beginners. Here are our criteria for the selection of books:The book should contain a variety of instructional materials, including exercises, examples, questions, learning activities, and other features that promote a programmer’s engagement and active learning.It must have a structured, clear, and logical progression of topics.Content must be up-to-date and should thoroughly teach and explain the basic concepts of C programming language.Use clear, precise, and easy-to-understand language.The book should have a clear layout and must be friendly toward self-taught programmers.Books make up the primary mode of learning. With so many books out there to learn C, the readers are left confused deciding which one to buy.Here are ten excellent C books for novice programmers:1. Best Book for Serious Learners: C: The Complete ReferenceC: The Complete Reference is another gem from Herbert Schildt and gives you full details on C99, the New ANSI/ISO Standard for C. You will get in-depth coverage of the C language and function libraries as well as all the newest C features, including restricted pointers, inline functions, variable-length arrays, and complex math. This jam-packed resource includes hundreds of examples and sample applications. The book gives complete descriptions of the entire C function library and provides tips on working efficiently in the C programming environment.The book provides insider tips, hundreds of examples, and expertly crafted explanations. The book is divided into 26 chapters and covers the following concepts:Data types and operatorsControl statementsClasses and objectsConstructors, destructors, and methodsInterfaces, arrays, enumerations, and structuresMethod and operator overloadingInheritance and virtual methodsReflection and runtime type IDException handlingDelegates, properties, events, and indexersAttributesMultithreading and the Task Parallel LibraryGenericsLINQ and PLINQLambda expressionsAnonymous typesExtension methodsNamed and optional argumentsImplicitly typed variablesI/O, networking, and collectionsThe book follows a structured approach to learning and answers all of the readers’ questions clearly and concisely.C in a Nutshell: The Definitive Reference by Peter Prinz and Tony Crawford helps you to learn how to build C programs with GNU and build executable programs from C source code.As you go through the book, you will be able to learn:C language concepts and language elements, with separate chapters on types, statements, pointers, memory management, I/O, and moreThe C standard library, including an overview of standard headers and a detailed function referenceBasic C programming tools in the GNU software collection, with instructions on how use them with the Eclipse IDEThe contents of the book are divided into three parts and twenty chapters. The first section covers all the important language elements like character, statements, points, memory management. The second section helps you to learn about the C standard library, and the third section teaches about the basic C programming tools.Part I LanguageChapter 1 covers Language BasicsChapter 2 covers TypesChapter 3 covers LiteralsChapter 4 covers Type ConversionsChapter 5 covers Expressions and OperatorsChapter 6 covers StatementsChapter 7 covers FunctionsChapter 8 covers ArraysChapter 9 covers PointersChapter 10 covers Structures, Unions and Bit-FieldsChapter 11 covers DeclarationsChapter 12 covers Dynamic Memory ManagementChapter 13 covers Input and OutputChapter 14 covers Processing DirectivesPart II Standard LibraryChapter 15 covers The Standard HeadersChapter 16 covers Functions at a GlanceChapter 17 covers Standard Library FunctionsPart III Basic ToolsChapter 18 covers Compiling with GCCChapter 19 talks about Using make to Build C ProgramsChapter 20 covers Debugging C Programs with GDBC Programming in easy steps by Mike McGrath has an easy-to-follow style that will appeal to anyone who wants to begin programming in C. The begins by explaining how to download and install a free C compiler so that you can quickly begin to create your own executable programs. Each chapter builds your knowledge of C.There are complete example programs that demonstrate each aspect of C together with screenshots that illustrate the output when that program has been executed. This book is ideal for novice programmers who have no previous knowledge of any programming language. The contents covered in the book are:Getting startedStoring variable valuesSetting constant valuesPerforming operationsMaking statementsEmploying functionsPointing to dataManipulating stringsBuilding structuresProducing resultsReference SectionBy the end of this book, you will have gained a sound understanding of the C language and be able to write your own C programs and compile them into executable files that can be run on any compatible computer.The new edition of Let Us C by Yashavant Kanetkar gives a detailed explanation of C programming language concepts.The book includes well-thought-out and fully working programming examples and exercises that would help you practice the skills learned in the chapter.The book follows an easy-to-follow style that appeals to everyone. It begins with the basics and steadily builds the pace so that the reader finds it easy to handle advanced topics towards the end of the book. The book is divided into twenty-three chapters.Chapter 1 covers Getting StartedChapter 2 covers C InstructionsChapter 3 covers Decision Control InstructionChapter 4 talks about More Complex Decision MakingChapter 5 covers Loop Control InstructionChapter 6 covers More Complex RepetitionsChapter 7 talks about Case Control InstructionChapter 8 talks about FunctionsChapter 9 covers PointersChapter 10 talks about RecursionChapter 11 covers Data Types RevisitedChapter 12 talks about The C PreprocessorChapter 13 covers ArraysChapter 14 talks about Multidimensional ArraysChapter 15 covers StringsChapter 16 covers Handling Multiple StringsChapter 17 covers StructuresChapter 18 talks about Console Input/OutputChapter 19 covers File Input/OutputChapter 20 talks about More Issues In Input/OutputChapter 21 covers Operations On BitsChapter 22 talks about Miscellaneous FeaturesChapter 23 covers Interview FAQsThis book is especially suited for someone who wants to begin programming in C. 5. Best Book for Theoretical Learners: Basic Computation and Programming with CBasic Computation and Programming with C is written by Subrata Saha and Subhodip Mukherjee for Beginner C programmers. It provides in-depth coverage of the fundamentals of computation and programming in C language. The book assists in the understanding of theoretical concepts and their applications.The unique approach of the book uses 'Learn by quiz’ including multiple-choice questions, with answers and explanations. It helps the reader to identify the right answer with adequate explanation and reasoning as to why the other options are incorrect. The book is divided into thirteen chapters and the contents covered in these chapters are:Chapter 1 covers Computer history, classification, and basic anatomyChapter 2 covers the Introduction to number system and logic gatesChapter 3 talks about the Introduction to system software and operating systemsChapter 4 talks about the Algorithms and flow chartChapter 5 covers the Introduction to CChapter 6 talks about Constants, variables, and data types Chapter 7 covers the Operators and expressionsChapter 8 covers the Input and output statementChapter 9 covers the Branching statementChapter 10 covers Loop statementsChapter 11 covers ArrayChapter 12 covers String handlingChapter 13 talks about FunctionChapter 14 covers PointerChapter 15 covers StructureChapter 16 talks about FileChapter 17 covers Bitwise operatorsChapter 18 talks about PreprocessorsChapter 19 talks about the Linked listThe book can be perfectly used as a self-learning book by beginners in Computer Programming.6. C Programming Absolute Beginner’s GuideC Programming Absolute Beginner’s Guide, written by Greg Perry and Dean Miller, teaches basic C language concepts using clear and easy-to-understand steps.The information on C programming language covered in this C book includes how to organize programs, store and display data, and work with variables, operators, I/O, pointers, arrays, and functions.It is the beginner’s guide to writing C programs. This programming book has also been updated for the new C11 standard and the free, open-source tools we have available today.Among these, you will also learn and discover:Free C programming for Windows, OS x, or LinuxHow to generate output and display it on the screenControl programs by testing dataUse logical operatorsStore easy-to-access informationUse functions to make programs easier to writeHow to handle your computer’s memory more efficiently Make programs more powerful using preprocessing directives7.Best for ANSI Standards: The C Programming Language (Second Edition)The C Programming Language Second Edition is a complete guide to ANSI standard C language programming. This programming book was written by C developers Brian W. Kernighan and Dennis M. Ritchie. The newest edition shows how to take advantage of operators, the economy of expression, improved control flow, and data structures offered by C programming.In this best C programming book, you will find helpful information, including:Detailed coverage of the C languageOfficial C language reference manualHelp with syntax notation, declarations, and scope rulesThe latest in ANSI changesAmong the content already listed, The C Programming Language book also covers other basic programming concepts, including variables, assignment statements, loops, and functions. The book is so easy to follow that even a novice programmer can follow along and pick up this programming language.Before reading this book, you should have fundamental programming knowledge, and it is not recommended for absolute beginners.8. Best C Book for Advancing Skills: Learn C the Hard WayLearn C the Hard Way by Zed A. Shaw offers practical exercises on the computational subjects you keep avoiding regarding C programming language. It is a good course for the beginner or junior programmers who are eager to advance their current skills in any language.In this book, you will find key concepts that cover:How to set up a C environmentBasic syntax and idiomsOperators, variables, and data typesProgram control structures and memory allocationData structuresDebugging and defensive codingBreaking and hacking your own C codeThe book begins with several crafted exercises and includes a teaching video. With this book, you will master defensive programming techniques, and work through practical projects to learn to apply the new concepts the book covers.9. Best Book for Complete Learning Experience: Head First CHead First C written by David Griffiths and Dawn Griffiths is a brain-friendly guide providing you with a complete learning experience that shows you how to create programs using C language. This book covers a more unique method that surpasses syntax and other how-to manuals. With this book, you learn how to be a great programmer.In addition to basic programming concepts, you will learn more about the following:Learn the secrets of C coding gurusLearn how to avoid mistakesPlay around in the C LibraryLearn how variadic functions can help you become more flexibleLearn how to use the compilerWith this book, you will also learn the latest in cognitive science and learning theory so you can craft a multi-sensory learning experience. The format of this book is described as visually rich and is not as text-heavy as some of the other C programming books we have covered.10.Best Book for More Advanced Students: Expert C Programming: Deep C SecretsExpert C Programming: Deep See Secrets was written by Peter Van Der Linden. This book is best suited for more advanced programmers who want a deeper understanding of more advanced tips and tricks. If you are a professional in computer fundamentals, this reference book can take your career to the next level.Linden uses a very conversational style in this book and uses examples that he has drawn from his own personal and practical experience. Each chapter of this reference book ends with a section titled, "Some Light Relief." This is where Linden discusses topics that some may find to be more recreational.This book will prove to be a valuable resource and includes information regarding:C programmer job interview secretsComputer software that blew up the space probe to VenusC code for a complete BASIC interpreterIt also includes an introduction to C++ that any programmer can understand, and it covers both IBM PC and UNIX systems. In addition to learning core concepts and the practical use of program execution, the reader will also be shown how to use this knowledge and have the opportunity to dive into the folklore of C.I hope you were able to find what you were looking for within these five best C programming books. I always emphasize the joy of learning programming languages above all. Only get these books if you know you will find utility and value from them.That said, the journey of programming languages does not stop with books.I always recommend pairing a book with a course for maximum immersion.Here are some courses on C programming:UdemyUdemy offers an extensive, highly-rated course, C Programming For BeginnersCourseraCoursera also partnered with Duke University to offer the Introduction to Programming in C Specialization. These are only a couple of great courses that will help you dive deep into C.If you’re on a budget, I have compiled over 70 free coding resources. It is a great collection. Many of the resources I have tried or used myself. If you’re interested, I look forward to seeing you in that article. ☺ If you are just starting out with your programming career and confused about which programming language to choose to learn to code then you are not alone. The sheer number of programming languages is simply overwhelming.It is, therefore, worth knowing that a good number of these languages heavily borrow their syntax from C, which makes starting with your programming journey with C Programming language, an apt choice.I have started my programming career with C and C++ before learning Java, and it’s been more than 11 years since I am using C++, though not as often as Java, which becomes my primary skill, my C++ knowledge has helped me to work on projects where both C++ and Java is used.Once considered the go-to language for general-purpose software development in the structured programming era, C has influenced quite a long list of languages to include PHP, JavaScript, Java, and Python.Using familiar C structures makes it possible for new languages to improve acceptance. C++ builds upon C to now offer support for objects and classes.Often time, I am rusty with C++ with very less coding, but whenever I have to refresh my knowledge, I have the right books and courses to do so. I have used C++ on both professional and personal projects and mostly use a GCC compiler.If you ask me which one is tough to learn, C++ or Java? I would say C++? It’s even difficult to master, and you cannot master it unless you are coding C++ daily.That’s why it’s also very important to choose the right books to learn from C++ from the start, but oftentimes it’s difficult given so many C++ books are available in the market.In this article, I’ll share a couple of books that I think are good for both beginners and intermediate C++ programmers. I consider myself in between a beginner and intermediate C++ programmer, as I actually code C++, use libraries from the Boost project, which is like JDK, but offer more algorithms than the Collection framework. By using Boost and STL, you feel like coding in Java because you don’t need to reimplement common functionalities like sockets, collections, serialization, etc. So, I have also included the best C++ books which explain STL and Boost, two essential libraries for any professional C++ developer. If you are a visual and active learner like me then you can also combine these books with an online course like Beginning C++ Programming — From Beginner to Beyond, which covers everything you need to know about CPP and also provides a more active and visual learning experience. This combination of a book and course works well to learn something new in a quick time.Without wasting any more of your time, here is a list of the best books to learn C and C++ from a beginner's point of view. Though, I have also included some advanced C++ books for experienced developers and senior students.1. C Programming Absolute Beginner’s GuideThis is one of the best C programming books a beginner should read. Taking into account that C has features and operations considered low level, and its different sized variables make it difficult for new programmers to learn.The approach used by the author gradually increases the complexity of the content after an extensive look at the basics. Using the Code: Blocks IDE, the book walks the reader through basic variable declarations to more advanced topics as working with functions and searching and sorting arrays.Authors: Greg Perry and Dean Miller Though the book is great for a beginner programmer, it would be a lot better if flow control was tackled much earlier in the book and some of the C related content such as heap management were left for the end. Nonetheless, the book is an effective guide to learning C and you can easily make up for those shortcomings by joining a C programming online course like C Programming for Beginners on Udemy. I highly suggest combining this book and course to start your C programming journey.This is probably the most recommend C programming book ever. Commonly referred to as the K&R, this book is a must-read book for all C programmers, including beginners and experienced developers.Authored by one of the designers of the C language, Dennis Ritchie, this short read is packed with the fundamental concepts behind the language. It is because of it that C became popularized during its early development. It served as the de facto standard for all who required a detailed C reference.Authors: Brian W Kernighan and Dennis M Ritchie The K&R covers the entire language’s specification which is quite rare for any other book in a different language. Other than the basics, C libraries are well covered and with numerous examples to try out. Its workflow is better organized and helps the learner to slowly build up on covered content.Here is the link to get this book — The C Programming Language (2nd Edition)If you need an online course to go with this book, I suggest you join Duke University’s Introduction to Programming in C specialization in Coursera. It’s a great collection of courses to learn and master C programming in a few weeks.Apart from that Coursera also has Coding for Everyone: C and C++ Specialization and Programming in C++: A Hands-on Introduction Specialization which are great to learn hands-on C++ coding and development By the way, instead of joining these courses and specialization individually, you can also join the Coursera Plus, a subscription plan from Coursera which gives you unlimited access to their most popular courses, specialization, professional certificate, and guided projects.This is one of the best books to learn C++ Programming from Scratch and it's good for both beginners and experienced programmers. Created by Bjarne Stroustrup this book is perfect to learn one of the most popular languages, C++ adds object-oriented concepts (data abstractions, polymorphism, inheritance, and encapsulation).Additionally, it introduces function overloading, and exception handling, which was previously absent in C programming. The language’s creator and developer authored this book to serve as documentation for the continuously evolving programming language. Consequent editions have been written to include changes in the language. Though it covers the language’s construct, some of its contents may appear advanced for novice programmers. However, with a good grasp of the basics, this book is excellent for all levels of programmers. Its companion book, David Vandevoorde’s C++ Solutions, offers learners step-by-step solutions to exercises found in Bjarne’s book.If you want, you can also combine this book with a free online course like C++ Tutorials for Beginners on Udemy to benefit from some active learning experience. This course is also completely free and more than 300K students have already joined it.Authors: Stanley B Lippman, Josee Lajoie, and Barbara E Moo It is agreed upon by many that this text forms one of the most effective introductions to the language. The book endeavors to be a reference book and a study tool for all levels of programmers.Only requiring a basic understanding of programming constructs, the book quickly builds upon one’s knowledge to cover important concepts such as C++’s template facility, algorithmic libraries, and classes. It provides great insights for even experienced programmers with its extensive coverage of the language’s features. As with the book’s tradition, the text continues to be kept up to date with the changing classifications of C++.This valuable resource also features a good number of new examples not included in older editions. If you need a course to go with this book, check out C++: From Beginner to Expert on Udemy.This C++ book is authored by Stephen Prata and it's awesome for both new programmers, students, and experienced developers who will find this book helpful in its coverage of C++’s programming logic.Its engaging explanations, coupled with well-explained examples, provide new programmers with a solid background in the language. The text also adequately covers object-oriented programming concepts. This is quite important as the topic is often quite confusing for new programmers. Farrell clarifies classes and objects making it possible for even new programmers to start developing more complex program structures. Similar to other texts in this list, the book thoroughly details the concept of templates as used in C++ as well as exception handling. Experienced programmers will find its explanation of random number generation informative especially for scientific simulations.And, if you need an advanced course to go with this book, check out Learn Advanced C++ Programming by John Purcell.I read Effective C++ by Scott Meyers after reading Effective Java by Joshua Bloch. As the name suggests, the book aims to teach you the right and effective way to use C++.This was written with the aim of being the second-best C++ programming book and it is right up there. This is a great book for all programmers coming from different backgrounds learning C++ like programmers coming from C and programmers coming from Java. The 3rd edition of Effective C++ contains 55 essays to improve your C++ program and design skills. This also helps to understand C++ better. In short, one of the must-read C++ books for experienced C++ developers.In short, one of the most must-read books for any C++ intermediate and experienced developer.Even though now C++ standard library has improved a lot in C++1, Boost is still the most used C++ library in both professional and personal projects. This book gives a good description of base Boost libraries. . This book is mainly for intermediate and experienced C++ programmers as it explains not only how to use boost library but also why you should use these libraries. As a professional C++ developer, it’s imperative to learn and get familiar with the Boost library, and there is no better book to do that than this one. You can also combine this book with the Learn Advanced C++ Programming course by John Purcell in Udemy. This course not only covers the C++ STL library but also introduces modern C++ concepts introduced in C++ 11 like Lambda expression.This is another gem from Scott Mayor, the most effective C++ author. This is basically the new version of Effective C++ to help C++ programmers upgrade themselves from C++03 to C++11 and C++14.I strongly suggest reading Effective Modern C++ along with Effective C++ to learn C++ better. If you need a course, you can also check out the C++ Standard Library including C++ 14 & C++ 17 course by Rainer Grimm on Educative for some interactive learning experience.This is one of the best C++ books from the C++ In-Depth Series. This book is about metaprogramming in C++, based on the Boost Metaprogramming Library. The first part of the book explains the basics of template programming and describes the boost MPL library quite well. It has a lot of examples to walk you through concepts, and it also explains metaprogramming, which many C++ programmers are not aware of. It’s not a beginner book but rather a good one for expert C++ programmers. If you need a course on C++ Template, I recommend the Beginner C++ Template course on Udemy. It’s a free course which means you don’t need to pay anything but you will learn a lot.This is one of the best books to learn Concurrency using C++, this book is authored by Anthony Williams and it’s a great book for experienced C++ developers or someone who wants to become a C++ expert programmer. It’s like what Java Concurrency in Action for Java developers. Concurrency is always been tricky in C++ and that’s the main area where Java scored well but C++ concurrency has come a long way. The first edition of this was an excellent book covering C++11 concurrency support including the thread library, the atomics library, the C++ memory model, locks, and mutexes, as well as issues of designing and debugging multithreaded and concurrent applications. This is the second edition which was released last year. It is updated to cover all the latest changes from C++ 14 and C++ 17. The book will teach you everything a C++ developer needs to know to write a robust, high-performance, and elegant concurrent C++ application. If you are serious about C++ programming I highly recommend you to read this book. If you want you can also combine this book with the Modern C++ Concurrency in Depth course by Kasun Liyanage on Udemy for quicker more active learning. It nicely complements this book.That’s all about the best books to learn C and C++ programming for beginners, students, junior programmers, and developers. Whether starting out as a hobbyist programmer or a computer science student, any of the books on the list would serve as an essential addition to your library.Even with new programming languages being developed every few years, C and C++ are still highly ranked. They have proved to be essential tools and a gateway to learning other programming languages.As such, they are still relevant and worth learning, even in today’s software development Other Programming Articles You May Like to ExploreP. S. — While these books are very good and perfect to learn both C and C++ but if it’s getting tougher for you then I also suggest you join a course like Beginning C++ Programming — From Beginner to Beyond, which covers everything you need to know about CPP and provides a more active and visual learning experience. Last updated on February 8th, 2025 at 12:33 pmC is the most basic programming language. It is easy to learn and quite user-friendly. Most people who want to learn to program, start with the C programming language. Are you one of them and looking for a good book on C programming?Deciding to learn the C programming language is an excellent decision as it is the base for all the programming languages. It doesn’t get easier than this. You can start here and then climb your way up.For your C programming journey, we have compiled a list of 15 books that are widely used by beginners as well as experts.Our top recommendations:This page contains some affiliate Links as we are participant in the Amazon Services LLC Associates Program. We may get compensation for each successful purchase through our link.You can learn the ANSI C programming language with this book. It will not only teach you C but will also help to transition to C++ and Java once you have grasped this language. The book starts by explaining the syntax of C. But it doesn’t stop there. The author goes beyond to help you understand the underlying logic of each syntax. You will find this book to be very sorted and clarified.Pros:All the material, right from abstract data types to function prototypes, is present in the book. You need not look anywhere else to master the C programming language.Cons:No Kindle version is available for this book.This book boasts of the fact that you need not have any prior experience in coding to learn C programming. In this book, you learn to organize programs, work with variables and operators, store and display data, pointers, arrays, I/O, functions, and much more. Whatever operating system you are using – OS X, Linux, or Windows, this book is for you.Pros:With consistent practice, you can code in C in no time with simply the help of this book.Cons:Some of the links to sample codes provided in the Kindle version do not work.This book is an introduction to professional C programming. It widely focuses on safe and secure systems. If you are worried about the security of your programs then definitely read the book thoroughly to understand the security aspects at each step of your code. This is an amazing guide if you see yourself building working systems and solving real-life problems with C programming.Pros:Each chapter contains practice code exercises so that you can get all the experience you need.Cons:This is not a book for absolute newbies to programming. You need to have some prior knowledge of coding.Who doesn’t like an easy way? This book gives you exactly that – an easy way to learn C programming. This book will create solid foundations for your programming journey. With this book, you will, of course, learn C programming but in conjunction with that, you will learn bits about software development. This makes this book ideal for beginners as well as experienced programmers.Pros:You will acquire skills that are not restricted to C language but are also applicable to other programming languages.Cons:There are errors in some of the sample codes which are not that difficult to spot but can get frustrating.The author follows the ANSI standard for C programming throughout the book. The book contains 8 chapters and 3 appendices. It covers most of the topics that any C programming book should have like control flow, functions and program structures, pointers and arrays, structures, input and output, the UNIX system interface, etc. The exercises at the end of each chapter make you think and use what you have learned to write a program all by yourself.Pros:This 2nd edition of the book has been updated to meet all the latest standards and requirements.Cons:You need to have a basic understanding of variables, assignment statements, loops, and functions in programming to start this book.Looking for a fully illustrated, colored book on C programming? You got it. The book uses very little jargon that makes it easy to understand and follow. The contents of this book are displayed in a very simplified manner making C less confusing. Even a child could learn C programming with this book. You get links to sample codes to check your work every step of the way as you proceed through this book.Pros:The book has been updated for the GNU Compiler version 6.3.0 and Windows 10.Cons:The book doesn’t contain any different information than most C programming books. The only thing that makes it stand out is the color-coded and illustrated content.Are you interested in learning socket programming in C? Then you should purchase this practical guide. When you are through, you will be able to write secure and optimized network code. You will basically get a grip on network protocols such as TCP and UDP and web protocols such as HTTP and HTTPS. You will gain experience with client-server applications in this book.Pros:The book is applicable for Windows, macOS, and Linux operating systems.Cons:The author has tried to cover many advanced topics in mere 614 pages. Due to that, enough explanation is lacking on most of the topics.The author stated that this is unlike any other C programming book on the market. He gives a lot of focus on pointers in programming. According to him, pointers give C its power and hence deserve to be learned in more depth than usually are. The author discusses tradeoffs between efficiency and maintainability at the end of most chapters. This book is for professionals and advanced students.Pros:The author provides programming tips and caution notes to get the best experience with C.Cons:No Kindle version is available for this book.The paperback version of this book is expensive.This is one of the most widely used books on C Programming. This book can be used as a textbook on C Programming as it has everything you will need to learn the language. The author starts with very basic elements of C and step-by-step takes you from simple to complex programs. The sample codes in this book are up-to-date and working. The end-of-the-chapter exercises make you a better programmer. And the elaborate notes at the end of each chapter help you recall all that you have learned.Pros:This book has everything you will ever need to start coding in C. You will not need any external resources other than those mentioned in the book.Cons:No Kindle version is available for this book.If you think every book on C you lay your hand on is not updated then you should choose this book. Updated with all the latest advancements in C programming this book serves to be an excellent tutorial for C. This book teaches you wonders like advanced math, talking to internet servers, running databases with existing C libraries, building high-level and object-based libraries, using modern syntactic functions, and a lot more.Pros:The author provides solid ideas for writing a clean code, gives good tips on efficient workflow, and teaches the use of excellent tools.Cons:The author assumes that you have prior knowledge of C Grammar and CS. If you don’t then this book will be a tough read.We just found you a definitive reference book for C programming. The book has dedicated chapters on concepts and elements in C language such as memory management, types, statements, pointers, I/O, etc. The latest edition of this book has information regarding features compliant with the C11 standard. Very easy-to-understand sections on GNU software collection will teach you to use various GNU tools.Pros:There is a clear distinction between the older C89/90 features and the newer C99/C11 revisions in the book.Cons:The index of this book is poorly prepared. It’s to navigate certain concepts.We present to you a practical handbook on C programming. Along with the mechanics of programming, this book will teach you to read, debug, and update programs with C. The author believes style and debugging are as important as syntax if not more. The most exciting thing about this book is the section on electronic archaeology which is the art of going through someone else’s code.Pros:The book covers Windows as well as Linux compilers.Cons:A beginner with very little experience in C will find this book useless as there are advanced concepts that need you to have a basic understanding of the language.Want to learn C programming in a fun way? There’s nothing better than this book. The author has maintained a humorous tone throughout the book which connects the reader very deeply with the material. The author uses stories, folklore, and anecdotes to explain the concepts of the C language. This book is not for beginners but rather for C programmers who want to expand their knowledge base.Pros:At the end of each chapter, you find a section titled ‘Dome Light Relief’ which discusses recreational topics related to C.Cons:The print quality of this book is very poor.A clear approach to C programming is followed in this book. The book is ideal for both students and professors. You will find coverage of both C89 and C99 standards. The author provides additional resources such as PowerPoint presentations of several concepts in the book and password-protected solutions and source codes to the exercises in the book. Our experts say that this is very easy to read and understood kind of book.Pros:There are almost 500 exercises and programming projects, in the book, for you to get hands-on experience.Cons:Several readers have found problems with the Kindle version of this book.Humans tend to learn more by example. This book adapts that technique to teach you C programming. Each new concept is illustrated by a C program. The author gives enough importance to good programming practices. Comprising 18 chapters and 5 appendices, this book is a complete package on C language. The end-of-the-chapter exercises are a great way to get practice in C programming.Pros:The author has included elegant yet complete examples of C that demonstrate a compilable piece of code.Cons:The Kindle edition of this book does not have page numbers which makes navigation difficult.C is said to be the easiest programming language. But that doesn’t mean it should be taken lightly as it is the foundation for other programming languages.If you are a complete beginner then take the help of the “C Programming- Absolute Beginner’s Guide” by Greg Perry and Dean Miller. It has everything you would like to learn.If you are someone who just can’t read a dull reference book but wants to learn C programming then we recommend purchasing “C Programming in Easy Steps” by Mike McGrath. This book is completely color-coded and illustrated with pictures.But our personal favorite is “The C Programming Language” by Brain W. Kernighan. May you be a beginner or an advanced reader you will not be disappointed with Mr. Kanetkar’s handy work. No matter whether you’re a working professional or a student, if you want to become a good software developer, your learning should never be stopped. But why just C or C++? The basic thing is that C provides programmers with what they desire, and they can design as per their needs. C++ was introduced as a successor to C, which enables programmers to use code with high mobility, and it also introduced the concept of objects and classes. In the present technology-driven world, these two have been most widely used among developers globally and are being adopted by all business scales, and that’s why the demand for C and C++ is booming every day.Yes, there are other languages too which you can learn, but the main reason to learn C is that it is the foundation to enter into the world of programming. However, one can master or learn these languages with the help of the best books that we’re going to discuss in this article today. Why Books? Reading enhances your ability to learn things at a higher pace. You get to know each concept in detail, which makes you a pro in that particular language. With the help of this article, we’ll understand the 10 best books on C and C++ for both beginners and professionals. Let’s find out.1. The C++ Programming Language (4th Edition) By Bjarne StroustrupThis book entails content for both beginners and professionals for mastering C++ language and was written by the creator of C++ itself i.e. Bjarne Stroustrup. This is the fourth version of the book which provides extensive details in a full-fledged form and also included exercises for practicing. With the help of this C++ book, you can learn the basic concepts of C++, libraries, containers, OOPS, and so on. The author has also contributed some other books based on C++ and currently, he’s serving as a member of the U.S National Academy of Engineering.2. C Programming Language (2nd Edition) By Brian W. Kernighan & Dennis M. RitchieThis 2nd edition book goes by ANSI (American National Standards Institute) standards and specifies the basic concept of C programming language. Written by Brian Kernighan and Dennis Ritchie, elaborates on the concept of OOPS and basic fundamentals of writing C language, and is considered among the best c programming book. This book is suitable for both beginner and advanced levels programmers and the best part about this book is one can learn all the concepts right from the beginning including functions and loops, libraries, data structure, etc. While practicing with this book, one can also solve different exercises like error correction, and modifications in the existing codes which will enhance conceptual understanding and knowledge.3. Head First C: A Brain-Friendly By David Griffiths and Dawn GriffithsEver wondered how you can master C with the help of just a book? With the help of this book “Head First C”, you can do wonders, no matter whether you’re a student or a working professional. This is among the best book for C that covers a wide range of methods and helps you to understand the language basics, pointers, dynamic memory management, etc. This book was released in 2012 by David Griffiths and Dawn Griffiths with the idea to share their expertise in a fun way and to make it more interesting they added a lab section with specific projects to test the reader’s ability and to develop top-notch skills.4. C++ Primer (5th Edition) By Stanley B. Lippman, Josee Lajoie, and Barbara E MooThis book is perfect for those who are looking to master their skill or learn about the C++ language. This book introduces all the libraries of C++ along with the functions. Since this is the fifth edition book, it comes with refreshed content for the new version of C++ and the examples you may find in this book is much more advanced and relevant for both advanced and beginners level of student or working professional. Written by Stanley Lippman, Josee Lajoie, and Barbara Moo, and was published back in 2012. This book can be a good choice for understanding C++ right from the root.5. Let us C By Yashavant P KanetkarTo learn the concept of C, this book is a one-stop solution for all. From describing the basic concept to going forward for in-depth knowledge, this book is suitable for everyone. Also considered as one of the best books for C, was written by Yashavant P Kanetkar who himself is a pioneer and has been awarded by Microsoft with “Most Valuable Profession” released his 17th edition of Let Us C in 2020. Even if someone is trying to prepare for competition, they can sharpen their skills by participating in various exercises and MCQs. This book covers all the basic details about arrays, functions, strings, etc., and has also a separate section for those who are preparing for job interviews.6. Programming: Principles and Practice Using C++ (2008) By Bjarne StroustrupThis is another masterpiece written by the creator of C++ himself in 2008. This book is well crafted for both beginners/advanced coders and as the reader will start diving in, it goes up to an advanced level. In this book, a student can go with the fundamental of C++, modern techniques of newer versions, libraries, and object-oriented programming for writing efficient codes. If you’ll go by the reviews only this is perhaps one of the best books, especially for those who have never done coding before. On the other hand, it also enhances the skillsets of those who know a bit of coding. The book is divided into different segments where a student will get to learn basic concepts, and techniques and the other phase will drive you with essential topics such as testing, text processing, etc., and also provides sufficient exercises for better understanding.7. C: The Complete Reference By Herbert SchildtThis is one of the best books for learning C for both beginner and advanced-level programmers. Written by Herbert Schildt has sold over 2.5 million copies worldwide so far and has been widely translated into different languages. The new edition gives an overview of arrays, functions, operators, expressions, structure, etc. as per ANSI standards. The insight of this book also covers the newest C libraries, variable-length arrays, inline functions, etc. For better understanding and brushing skillset, it comes with a variety of examples and exercises at the end of every unit which you can solve to master in C language. 8. Effective Modern C++ (2014) By Scott MeyersThose who are looking to excel in their skills in C++ should definitely consider getting this book by Scott Meyers. It was introduced to improve the concept of C++11 and C++ 14 and ways of implementing them in a real-life project. This book covers the basic concept of C++ which consists of lambda expressions, type declarations, etc. Apart from this, the user will be also able to learn how to develop an efficient coding technique using C++. It has a vast variety of subjects that have been used for better understanding like the pros and cons of braced initialization, the relationship between std:: move, std:: forward, and also explains how std:: atomic is different from volatile. So, in contrast, if you’ve started learning C++ and want to learn how you can implement coding in real-life projects then you might seek the help of Effective Modern C++ for a better understanding.9. Programming in ANSI C By E. BalaguruswamyThis book has been designed for every group of programmers i.e. beginners to advanced and has been crafted well and covers every level of this mother language of programming i.e. C. Written by E. Balaguruswamy kept this in mind how it will reflect users who have no prior experience in coding and that's the reason why it has been listed among the Best C Programming Books. Besides this, the newer edition of this c programming book covers the latest functions, arrays, strings, etc. The in-depth covers all the exercise parts which programmers can utilize to suit their skillset. The 8th edition covers the updated content as per AICTE standards for students with strong pedagogical features, graphic programming using C, and much more.10. Practical C++ Programming (2nd Edition) By Steve OuallineThe second edition of practical C++ programming offers a wide range of assessments, redefined introduction, and a brief history of C++ and also explains how to design anything with C++ with perfection. This book covers C++ syntax coding, style creation, use of object classes, template debugging, etc. Those who want to learn C++ can take the help of this book and focus on the introduction part and ensure to complete all the exercises that have been provided at the end of every unit. By the end of the book, it will show you how you can implement it in real-life projects by writing clean and efficient code.
ConclusionPerhaps you might find ample of c programming books on different resources but only a handful of them are offering the best variety of matter for both beginners and advanced-level programmers. We tried to pull out the list of 10 Best C and C++ Books For Beginners & Advanced Programmers that are worth giving a try and getting your dream job in your core specialty such as C++ Analyst, Junior/Senior Programmer, Game Programmer, etc.Must Read Computer software installed on multiple computing platforms "Cross-platform" redirects here. For the railway station interchange, see cross-platform interchange. For the game term, see cross-platform play. "Multi-platform" redirects here. For the mode of storytelling in television, see multi-platform television. Within computing, cross-platform software (also called multi-platform software, platform-agnostic software, or platform-independent software) is computer software that is designed to work in several computing platforms.[1] Some cross-platform software requires a separate build for each platform, but some can be directly run on any platform without special preparation, being written in an interpreted language or compiled to portable bytecode for which the interpreters or run-time packages are common or standard components of all supported platforms.[2] For example, a cross-platform application may run on Linux, macOS and Microsoft Windows. Cross-platform software may run on many platforms, or as few as two. Some frameworks for cross-platform development are Codename One, ArkUI-X, Kivy, Qt, GTK, Flutter, NativeScript, Xamarin, Apache Cordova, Ionic, and React Native.[3] Main article: Computing platform Platform can refer to the type of processor (CPU) or other hardware on which an operating system (OS) or application runs, the type of OS, or a combination of the two.[4] An example of a common platform is Android which runs on the ARM architecture family. Other well-known platforms are Linux/Unix, macOS and Windows, these are all cross-platform.[4] Applications can be written to depend on the features of a particular platform—either the hardware, OS, or virtual machine (VM) it runs on. For example, the Java platform is a common VM platform which runs on many OSs and hardware types. A hardware platform can refer to an instruction set architecture. For example: ARM or the x86 architecture. These machines can run different operating systems. Smartphones and tablets generally run ARM architecture, these often run Android or iOS and other mobile operating systems. A software platform can be either an operating system (OS) or programming environment, though more commonly it is a combination of both. An exception is Java, which uses an OS-independent virtual machine (VM) to execute Java bytecode. Some software platforms are: Android (ARM64) ChromeOS (ARM32, ARM64, IA-32, x86-64) Common Language Infrastructure (CLI) by Microsoft, implemented in: The legacy .NET Framework that works only on Microsoft Windows. The newer .NET framework (simply called ".NET") that works across Microsoft Windows, macOS, and Linux. Other implementations such as Mono (formerly by Novell and Xamarin[5]) HarmonyOS (ARM64, RISC-V, x86, x64,

manual effort for testing and maintenance.[7] Techniques such as full virtualization are sometimes used as a workaround for this problem. Tools such as the Page Object Model allow cross-platform tests to be scripted so that one test case covers multiple versions of an app. If different versions have similar user interfaces, all can be tested with one test case. Web applications are becoming increasingly popular but many computer users still use traditional application software which does not rely on a client/web-server architecture. The distinction between traditional and web applications is not always clear. Features, installation methods and architectures for web and traditional applications overlap and blur the distinction. Nevertheless, this simplifying distinction is a common and useful generalization. Traditional application software has been distributed as binary files, especially executable files. Executables only support the platform they were built for—which means that a single cross-platform executable could be very bloated with code that never executes on a particular platform. Instead, generally there is a selection of executables, each built for one platform. For software that is distributed as a binary executable, such as that written in C or C++, there must be a software build for each platform, using a toolset that translates—transcompiles—a single codebase into multiple binary executables. For example, Firefox, an open-source web browser, is available on Windows, macOS (both PowerPC and x86 through what Apple Inc. calls a Universal binary), Linux, and BSD on multiple computer architectures. The four platforms (in this case, Windows, macOS, Linux, and BSD) are separate executable distributions, although they come largely from the same source code. In rare cases, executable code built for several platforms is combined into a single executable file called a fat binary. The use of different toolsets may not be enough to build a working executables for different platforms. In this case, programmers must port the source code to the new platform. For example, an application such as Firefox, which already runs on Windows on the x86 family, can be modified and re-built to run on Linux on the x86 (and potentially other architectures) as well. The multiple versions of the code may be stored as separate codebases, or merged into one codebase. An alternative to porting is cross-platform virtualization, where applications compiled for one platform can run on another without modification of the source code or binaries. As an example, Apple's Rosetta, which is built into Intel-based Macintosh computers, runs applications compiled for the previous generation of Macs that used PowerPC CPUs. Another example is IBM PowerVM Lx86, which allows Linux/x86 applications to run unmodified on the Linux/Power OS. Example of cross-platform binary software: The LibreOffice office suite is built for Microsoft Windows, macOS, Linux, FreeBSD, NetBSD, OpenBSD, Android, iOS, iPadOS, ChromeOS, web-based Collabora Online and many others.[8][9] Many of these are supported on several hardware platforms with processor architectures including IA-32, x86-64, ARM (ARMel, ARMhf, ARM64), MIPS, MIPSel, PowerPC, ppc64le, and S390x[9][10] A script can be considered to be cross-platform if its interpreter is available on multiple platforms and the script only uses the facilities built into the language. For example, a script written in Python for a Unix-like system will likely run with little or no modification on Windows, because Python also runs on Windows; indeed there are many implementations (e.g. IronPython for .NET Framework). The same goes for many of the open-source scripting languages. Unlike binary executable files, the same script can be used on all computers that have software to interpret the script. This is because the script is generally stored in plain text in a text file. There may be some trivial issues, such as the representation of a new line character. Some popular cross-platform scripting languages are: bash – A Unix shell commonly run on Linux and other modern Unix-like systems, as well as on Windows via the Cygwin POSIX compatibility layer, Git for Windows, or the Windows Subsystem for Linux. Perl – First released in 1987. Used for CGI programming, small system administration tasks, and more. PHP – Mostly used for web applications. Python – A language which focuses on rapid application development and ease of writing, instead of run-time efficiency. Ruby – An object-oriented language which aims to be easy to read. Can also be used on the web through Ruby on Rails. Tcl – A dynamic programming language, suitable for a wide range of uses, including web and desktop applications, networking, administration, testing and many more. Cross-platform or multi-platform is a term that can also apply to video games released on a range of video game consoles. Examples of cross-platform games include: Miner 2049er, Tomb Raider: Legend, FIFA series, NHL series and Minecraft. Each has been released across a variety of gaming platforms, such as the Wii, PlayStation 3, Xbox 360, personal computers, and mobile devices. Some platforms are harder to write for than others, requiring more time to develop the video game to the same standard. To offset this, a video game may be released on a few platforms first, then later on others. Typically, this happens when a new gaming system is released, because video game developers need to acquaint themselves with its hardware and software. Some games may not be cross-platform because of licensing agreements between developers and video game console manufacturers that limit development to one particular console. As an example, Disney could create a game with the intention of release on the latest Nintendo and Sony game consoles. Should Disney license the game with Sony first, it may be required to release the game solely on Sony's console for a short time or indefinitely. Main articles: Cross-platform play and List of video games that support cross-platform play Several developers have implemented ways to play games online while using different platforms. Psyonix, Epic Games, Microsoft, and Valve all possess technology that allows Xbox 360 and PlayStation 3 gamers to play with PC gamers, leaving the decision of which platform to use to consumers. The first game to allow this level of interactivity between PC and console games (Dreamcast with specially produced keyboard and mouse) was Quake 3.[11][12] Games that feature cross-platform online play include Rocket League, Final Fantasy XIV, Street Fighter V, Killer Instinct, Paragon and Fable Fortune, and Minecraft with its Better Together update on Windows 10, VR editions, Pocket Edition and Xbox One. Cross-platform programming is the practice of deliberately writing software to work on more than one platform. There are different ways to write a cross-platform application. One approach is to create multiple versions of the same software in different source trees—in other words, the Microsoft Windows version of an application might have one set of source code files and the Macintosh version another, while a FOSS *nix system might have a third. While this is straightforward, compared to developing for only one platform it can cost much more to pay a larger team or release products more slowly. It can also result in more bugs to be tracked and fixed. Another approach is to use software that hides the differences between the platforms. This abstraction layer insulates the application from the platform. Such applications are platform agnostic. Applications that run on the JVM are built this way. Some applications mix various methods of cross-platform programming to create the final application. An example is the Firefox web browser, which uses abstraction to build some of the lower-level components, with separate source subtrees for implementing platform-specific features (like the GUI), and the implementation of more than one scripting language to ease software portability. Firefox implements XUL, CSS and JavaScript for extending the browser, in addition to classic Netscape-style browser plugins. Much of the browser itself is written in XUL, CSS, and JavaScript. There are many tools[13][14] available to help the process of cross-platform programming: 8th: a development language which utilizes Juce as its GUI layer. It currently supports Android, iOS, Windows, macOS, Linux and Raspberry Pi. Anant Computing: A mobile application platform that works in all Indian languages, including their keyboards, and also supports AppWallet and native performance in all OSs. AppearIQ: a framework that supports the workflow of app development and deployment in an enterprise environment. Natively developed containers present hardware features of the mobile devices or tablets through an API to HTML5 code thus facilitating the development of mobile apps that run on different platforms. Boden: a UI framework written in C++. Cairo: a free software library used to provide a vector graphics-based, device-independent API. It is designed to provide primitives for 2-dimensional drawing across a number of different backends. Cairo is written in C and has bindings for many programming languages. Cocos2d: an open-source toolkit and game engine for developing 2D and simple 3D cross-platform games and applications. Codename One: an open-source Write Once Run Anywhere (WORA) framework for Java and Kotlin developers. Delphi: an IDE which uses a Pascal-based language for development. It supports Android, iOS, Windows, macOS, Linux. Ecere SDK: a GUI and 2D/3D graphics toolkit and IDE, written in eC and with support for additional languages such as C and Python. It supports Linux, FreeBSD, Windows, Android, macOS and the Web through Emscripten or Binaryen [Wikidata] (WebAssembly). Eclipse: an open-source development environment. Implemented in Java with a configurable architecture which supports many tools for software development. Add-ons are available for several languages, including Java and C++. FLTK: an open-source toolkit, but more lightweight because it restricts itself to the GUI. Flutter: A cross-platform UI framework for IOS, Android, Mac, Windows and developed by Google. fpGUI: An open-source widget toolkit that is completely implemented in Object Pascal. It currently supports Linux, Windows and a bit of Windows CE. GeneXus: A Windows rapid software development solution for cross-platform application creation and deployment based on knowledge representation and supporting C#, COBOL, Java including Android and BlackBerry smart devices, Objective-C for Apple mobile devices, RPG, Ruby, Visual Basic, and Visual FoxPro. GLBasic: A BASIC dialect and compiler that generates C++ code. It includes cross compilers for many platforms and supports numerous platform (Windows, Mac, Linux, Android, iOS and some exotic handhelds). Godot: an SDK which uses Godot Engine. GTK+: An open-source widget toolkit for Unix-like systems with X11 and Microsoft Windows. Haxe: An open-source language. Juce: An application framework written in C++, used to write native software on numerous systems (Microsoft Windows, POSIX, macOS), with no change to the code. Kivy: an open-source cross-platform UI framework written in Python. It supports Android, iOS, Linux, OS X, Windows and Raspberry Pi. LEADTOOLS: Cross-platform SDK libraries to integrate recognition, document, medical, imaging, and multimedia technologies into Windows, iOS, macOS, Android, Linux and web applications.[15] LiveCode: a commercial cross-platform rapid application development language inspired by HyperTalk. Lazarus: A programming environment for the FreePascal Compiler. It supports the creation of self-standing graphical and console applications and runs on Linux, MacOSX, iOS, Android, WinCE, Windows and WEB. Max/MSP: A visual programming language that encapsulates platform-independent code with a platform-specific runtime environment into applications for macOS and Windows A cross-platform Android runtime. It allows unmodified Android apps to run natively on iOS and macOS Mendix: a cloud-based low-code application development platform. MonoCross: an open-source model–view–controller design pattern where the model and controller are cross-platform but the view is platform-specific.[16] Mono: An open-source cross-platform version of Microsoft .NET (a framework for applications and programming languages) MoSync: an open-source SDK for mobile platform app development in the C++ family. Mozilla application framework: an open-source platform for building macOS, Windows and Linux applications. OpenGL: a 3D graphics library. Pixel Game Maker MV: A proprietary 2D game development software for Windows for developing Windows and Nintendo Switch games. PureBasic: a proprietary language and IDE for building macOS, Windows and Linux applications. ReNative: The universal development SDK to build multi-platform projects with React Native. Includes latest iOS, tvOS, Android, Android TV, Web, Tizen TV, Tizen Watch, LG webOS, macOS/OSX, Windows, KaiOS, Firefox OS and Firefox TV platforms. Qt: an application framework and widget toolkit for Unix-like systems with X11, Microsoft Windows, macOS, and other systems—available under both proprietary and open-source licenses. Simple and Fast Multimedia Library: A multimedia C++ API that provides low and high level access to graphics, input, audio, etc. Simple DirectMedia Layer: an open-source multimedia library written in C that creates an abstraction over various platforms' graphics, sound, and input APIs. It runs on OSs including Linux, Windows and macOS and is aimed at games and multimedia applications. Smartface: a native app development tool to create mobile applications for Android and iOS, using WYSIWYG design editor with JavaScript code editor. Tcl/Tk Titanium Mobile: open source cross-platform framework for Android and iOS development. U++: a C++ GUI framework for performance. It includes a set of libraries (GUI, SQL, etc..), and IDE. It supports Windows, macOS and Linux. Unity: Another cross-platform SDK which uses Unity Engine. Uno Platform: Windows, macOS, iOS, Android, WebAssembly and Linux using C#. Unreal: A cross-platform SDK which uses Unreal Engine. V-Play Engine: V-Play is a cross-platform development SDK based on the popular Qt framework. V-Play apps and games are created within Qt Creator. WaveMaker: A low-code development tool to create responsive web and hybrid mobile (Android & iOS) applications. WinDev: an Integrated Development Environment for Windows, Linux, .Net and Java, and web browers. Optimized for business and industrial applications. wxWidgets: an open-source widget toolkit that is also an application framework.[17] It runs on Unix-like systems with X11, Microsoft Windows and macOS. Xojo: a RAD IDE that uses an object-oriented programming language to compile desktop, web and iOS apps. Xojo supports natively compiling to Windows, macOS, iOS and Linux, and can also create compiled web apps that are able to be run as standalone servers or through CGI. This section possibly contains original research. Please improve it by verifying the claims made and adding inline citations. Statements consisting only of original research should be removed. (March 2025) (Learn how and when to remove this message) There are many challenges when developing cross-platform software: Testing cross-platform applications may be considerably more complicated, since different platforms can exhibit slightly different behaviors or subtle bugs. This problem has led some developers to deride cross-platform development as "write once, debug everywhere", a take on Sun Microsystems' "write once, run anywhere" marketing slogan. Developers are often restricted to using the lowest common denominator subset of features which are available on all platforms. This may hinder the application's performance or prohibit developers from using the most advanced features of each platform. Different platforms often have different user interface conventions, which cross-platform applications do not always accommodate. For example, applications developed for macOS and GNOME are supposed to place the most important button on the right-hand side of a window or dialog, whereas Microsoft Windows and KDE have the opposite convention. Though many of these differences are subtle, a cross-platform application which does not conform to these conventions may feel clunky or alien to the user. When working quickly, such opposing conventions may even result in data loss, such as in a dialog box confirming whether to save or discard changes. Scripting languages and VM bytecode must be translated into native executable code each time they are used, imposing a performance penalty. This penalty can be alleviated using techniques like just-in-time compilation; but some computational overhead may be unavoidable. Different platforms require the use of native package formats such as RPM and MSI. Multi-platform installers such as InstallAnywhere address this need. Cross-platform execution environments may suffer cross-platform security flaws, creating a fertile environment for cross-platform malware.[18] Operating context List of widget toolkits Hardware virtualization Language binding Source-to-source compiler Binary-code compatibility Comparison of user features of messaging platforms

^ "Design Guidelines: Glossary". java.sun.com. Archived from the original on 2012-02-13. Retrieved 2011-10-19. ^ "SDD Technology blog: Definition of cross platform". SDD Technology. Retrieved 2020-10-18. ^ Lee P Richardson (2016-02-16). "Xamarin vs Ionic: A Mobile, Cross Platform, Shootout". ^ a b "Platform Definition". The Linux Information Project. Retrieved 2014-03-27. ^ "About Mono". mono-project.com. Retrieved 2015-12-17. ^ Corti, Sascha P. (October 2011). "Browser and Feature Detection". MSDN Magazine. Retrieved 28 January 2014. ^ Choudhary, S.R. (2014). "Cross-platform testing and maintenance of web and mobile applications". Companion Proceedings of the 36th International Conference on Software Engineering. pp. 642–645. doi:10.1145/2591062.2591097. hdl:1853/53588. ISBN 9781450327688. S2CID 1903037. ^ Mehrotra, Pranob (2020-12-01). "Collabora Office suite gets a new layout for Android tablets and Chromebooks". XDA-Developers. Retrieved 2021-01-15. Collabora Office is a popular open-source alternative to the Microsoft Office suite. It's based on LibreOffice, and it's available on a variety of platforms, including Windows, Linux, iOS, and Android. This year in July, a major update for the office suite brought support for Chrome OS devices. ^ "Collabora Office on iOS and Android Just got Better!". Adfinis. 2020-12-15. Retrieved 2021-01-15. ...touch optimized interfaces: one for tablets and one for phone screens. ...(iOS, iPadOS, Chromebooks, Android). ^ "Nextcloud Ubuntu Appliance adds Collabora Online to Raspberry Pi image". MuyLinux. 2021-03-26. Retrieved 2021-03-30. the first viable self-hosted web office solution for the popular Raspberry Pi 4 platform ^ Cribba. Quake III Arena, Giant Bombcast, February 15, 2013. ^ A Closer Look At The Dreamcast Internet Starter Kit ^ The GUI Toolkit, Framework Page ^ "Platform Independent FAQ". Archived from the original on 2008-08-16. Retrieved 2009-04-25. ^ "Cross-Platform SDK Libraries for Recognition, Document, Medical, Imaging, and Multimedia". www.leadtools.com. Retrieved 2021-03-03. ^ "12 benefits of Xamarin Cross-platform app development". HeadWorks. 15 Mar 2019. ^ WxWidgets Description ^ Warren, Tom (2020-01-14). "Microsoft bids farewell to Windows 7 and the millions of PCs that still run it". The Verge. Retrieved 2020-02-06. Retrieved from "