IJRESM
International Journal of Research in Engineering, Science and Management
Volume-2, Issue-5, May-2019
www.ijresm.com | ISSN (Online): 2581-5792

Application of C/C++ Compiler

Nikhil Deshpande[1], Sameer Joshi[2], Sagar Pawar[3], A. Parkavi[4]
[1,2,3]Student, Dept. of Computer Science and Engg., M. S, Ramaiah Institute of Technology, Bengaluru, India
[4]Assistant Professor, Dept. of Computer Science and Engg., M. S, Ramaiah Inst. of Tech., Bengaluru, India

***Abstract*: A compiler changes and converts a code written in a particular source language into an equivalent language through a so many of steps. A new approach GLAP model for time and design complexity analysis of lexical analyzer is proposed in this paper [1]. Digital signal processors have been developed which performs 32-bit floating-point operations at a rate of 25 speeds and can be coded using a standard C compiler [2]. The paper mainly aims to explain an online C compiler which helps to reduce the problems and provides solution of portability and storage space by making use of the cloud computing [3]. The paper proposes a C source compiler able to increase the dependability properties of a given application. The adopted strategies are based on two main and major techniques, variables duplication and code re-ordering [4]. As we work with cross platform and mobility constraints, traditional C compilers fail to work along. "Cross Platform C Compiler (CPCC)" is a Web- based application which aims to overcome these limitations. Unlike other online compilers, CPCC uses client-side scripting for execution of codes, which reduces the computational load on the server side [5].**

A C compiler-based methodology for a given common task of audio application from the C code to executable image which targets an audio fixed-point to mid-scale using DSPs [6]. A C compiler for TMS320C50 is very convenient and efficient for the compiling C programs, TMS320C50 supports the 'fix' data type [7]. Digital Signal Processors are used in real time applications for compiling C programs [8]. Buffer overflow it's the main feature of software security, it is used to reduce the threat and to randomize the address of the given compiled C program using compiler [9]. The performance of different programming language such as C, C++ and JAVA based on OS/390 [10] is measured in this paper. We briefly describe the performance of the languages generated by the code generated. There are two methods of comparison of benchmark i.e. On same platform and different platform to overcome the limitations and advantages different standards of various platform.

***Keywords*: C/C++ Compiler**

1. Introduction

Operating systems and Compilers constitute the basic interfaces between a programmer and the machine. The term compilation denotes the conversion of an algorithm expressed in a human-oriented source language to an equivalent algorithm expressed in a hardware-oriented target language [1] Digital signal processors have been developed which performs 32-bit floating-point operations at a rate of 25 speeds and can be coded using a standard C compiler.

The processor is based on the architecture of the DSP32 which is introduced in 1980s and has found wide application in numeric processing, telecommunications, control systems and computer graphics [2]. Each compiler majorly consists of front end to check semantics, middle end which performs the optimization through removal or useless code, back end where translation takes place. Cloud computing is used to build compiler online [3]. This paper talks about a pure software and fully automated approach base on variable duplication and code re-ordering to detect data errors appearing in the system memory. Source C compiler has been in order to transform any input source code into an output reliable code, properly modified to increase its dependability characteristics [4]. A compiler converts a program written in a high-level programming language. Using a high-level language for programming has a huge impact on how quick programs can be developed. CPCC incorporates this mechanism of code execution by converting the C codes into their equivalent JavaScript codes which are executable codes for browsers [5]. This paper gives the methodology for implementation process using DSPs. It uses the principle of Cirrus Logic Coyote [6]. High level languages which is required for programming of digital signal processors by using TMS320C50 [7]. Digital Signal Processors is been improved a lot in the recent years, DSPs are being used to implement the algorithms for example DTMF receiver [8]. Buffer overflow it's the main feature of software security, it is used to reduce the threat and to randomize the address of the given compiled C program using compiler [9].

Compilers are building blocks of any giant software including embedded or enterprise system. We may have many well developed or mature compilers such as GCC and LLVM these compilers work with a very high efficiency and fewer errors when it comes to traditional computation sometimes they may not handle the bugs or error created by new techniques, therefore many bugs are fixed and updated on the daily basis. Generally, the Compilers are flooded with huge number of tests it is quite difficult to validate any compliers based on the tests flooded. There have been many attempts to give out the best random test system till date Csmith stands out to be the best it was able to report several bugs in each GCC and LLVMs which are amongst the well-established compilers we have.

2. Literature survey

For a C Compiler, it is not difficult to see that this translation process from source text to instruction sequence requires considerable effort and follows complex rules. The construction of first C compiler involved about 18-man effort and figured among the largest programming projects of the time [1]. The main advantage of cloud computing over the other non-network methods is quickly processing. The main reason for creating the project is to provide a centralized compiling scheme for organizations or institutions. Also, the process of maintenance and distribution of dynamic usernames and passwords will be greatly simplified [3]. Many Researchers showed high percentage of faults cause, A Fail Silent Violation Behaviour, it is said that it is possible to achieve a high degree of Fair Silent Behaviour in ordinary computers [4]. A compiler generates an object file for compilation of the host alone. The cross compiler switches it configuration from the host specific to the embedded system specific. The client gets the connection of all the executable records that are available, and the code is then executed [5].

The C compiler for the embedded Systems it compares the implementation flow steps with one another in order to get the required output for the given program. It also checks the code using DSPs which is based on embedded Systems [6]. The TMS320C50 structure uses the GNU C compiler for the fixed-point C programs. The compiler structure uses the scaling and optimization code [7]. The DSP32 contains 160ns instructions which can be divided into cycle times, and it can execute over

12.5 million of different floating-point operations per second for the given program [8]. Visual Studio C++ which helps in reducing the attackers which can attack on over-flow and cookie. If a program runs in the Visual Studio C++ with GS option on, it produces a cookie which may lead to security issues of the system [9].

C is the most preferred language by the beginners hence it makes the cross compiler platform necessary (CPCC). There are different modules present in this system with different functionality some of the modules are Registration Module, Code Selection Module Code Storage Module, Challenges, Execution, Evaluation this make a complete package of cross compiler platform. CPCC is mainly designed for students, educational bodies. Students login their credentials to exploit the features. This Platform is quite user friendly admins can post or withdraw comments on the codes present on the platform, admins are also given the veto power to view the users account to maintain the platforms accessibility.

3. Conclusion

Simplicity and Modularity of the code are important for reasons such as your code is much more likely to be correct, and you will be able to respond to changes in the source language specification. In this course, it is set very lax targets for both, emphasizing correctness instead. This will lead to optimise the generated code even more [1]. A new digital signal processor called DSP32C is used to perform 32-bit floating point operations at certain speed and can be programmed using C compiler. The speed of this can be even more increased. Hence this paper concludes by saying about the increase in the speed of Compiler [2]. By integrating and enhancing the capabilities of these essential technologies, here it is introduced that of 'Online Compiler' and to contribute to the current examination system. There would be a cloud where there will be a server which would have the power to compile the students code stored on another machine [3]. Here RECCO, Reliable Code Compiler, able to automatically generate reliable version of any source C code. The tool exploits an effective code reordering algorithm and a customizable variable duplication technique to generate a reliable code able to detect the occurrence of critical data faults [4]. Here it is thus analyzed the traditional C compilers and identified the major limitations as platform dependency and immobility of codes. So we have proposed CPCC (Cross Platform C Compiler); a Web Application; which makes the C compiler platform independent. Besides this, it is faster than Turbo C which helps programmers to complete their work swiftly [5]. It gives the amount of time consumed by a man to execute a given program within the given period along with their code size. This application tries to match the implementation complexity which compares the developer and the code which is written by them [6]. This paper gives the new data types which supports fixed-point arithmetic rules using TMS320C50 which gives digital signal processor. It provides the C compiler which checks the acceptable comparison for the SQNR [7]. The given paper provides the simple set of rules for generating assembly language code which is present in compiler. It uses DSP32/C to the given set of instructions in the system by considering their optimization strategies [8]. Visual studio provides the security check tools to avoid the risk of vulnerable buffers. The drawback of the Visual studio is that they cannot prevent all the kinds of buffer overflows. Visual studio is used to increase the security of the software and to avoid risk of hacking by some third parties [9].

Like most of the compilers architecture this compiler is also divided into three parts Front end ,the Middle end and the Back end .All the parts have their respective jobs initiating with Front end it checks for syntactical and lexical errors after handling this errors it converts the given code into some intermediate representation (IR).Everything is in the form of tree this in technical term is called as GENERIC[10]. In the middle layer the major part of the optimizing the complex statement is done by GIMPLE [11]. Conversion of code form from generic to gssimple is called gimplification . Further in Back end the code received from the GIMPLE is optimized and the intermediate representation(IR) is converted into Ma- chine code and feed into machines.

References

[1] Jatin Chhabra, Hiteshi Chopra, and Abhimanyu Vats, "Research paper on Compiler Design," in Internatonal Journal of Innovative Research in Technology, vol. 1, no. 5, 2014, pp. 151-153.

[2] J.R. Boddie; C.J. Garen; M.L. Fuccio; J. Tow, A floating point DSP with optimizing C compiler, 1988 International Conference onA coustics, Speech, and Signal Processing, 1988. ICASSP-88., May 1988.

[3] Aamir Nizam Ansari, S. Patil, A. Navada, A. Peshave and V. Borole, "Online C/C++ compiler using cloud computing," *2011 International Conference on Multimedia Technology*, Hangzhou, 2011, pp. 3591-3594.

[4] A. Benso, S. Chiusano, P. Prinetto and L. Tagliaferri, "A C/C++ source-to-source compiler for dependable applications," *Proceeding International Conference on Dependable Systems and Networks. DSN 2000*, New York, NY, USA, 2000, pp. 71-78.

[5] F. Mulla, S. Nair and A. Chhabria, "Cross Platform C Compiler," *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*, Pune, 2016, pp. 1-4.

[6] M. Djukic, N. Cetic, J. Kovacevic and M. Popovic, "A C compiler based methodology for implementing audio DSP applications on a class of embedded systems," *2008 IEEE International Symposium on Consumer Electronics*, Vilamoura, 2008, pp. 1-4.

[7] Jayang Kang and Wonyong Sung, "Fixed-point C compiler for TMS320C50 digital signal processor," *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Munich, 1997, pp. 707-710 vol.1.

[8] J. Hartung, S. L. Gay and S. G. Haigh, "A practical C language compiler/optimizer for real-time implementations on a family of floating point DSPs," *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, New York, NY, USA, 1988, pp. 1674-1677 vol.3.

[9] Y. WU, "Enhancing Security Check in Visual Studio C/C++ Compiler," *2009 WRI World Congress on Software Engineering*, Xiamen, 2009, pp. 109-113.

[10] D. A. Cargill and M. Radaideh, "A practitioner report on the evaluation of the performance of the C, C++ and Java compilers on the OS/390 platform," *2000 IEEE International Symposium on Performance Analysis of Systems and Software. ISPASS (Cat. No.00EX422)*, Austin, TX, USA, 2000, pp. 40-45.

[11] D. Bokan, M. Đukić, M. Popović and N. Četić, "Adjustment of GCC compiler frontend for embedded processors," *2014 22nd Telecommunications Forum Telfor (TELFOR)*, Belgrade, 2014, pp. 983-986.

[12] K. Nakamura and N. Ishiura, "Random testing of C compilers based on test program generation by equivalence transformation," *2016 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, Jeju, 2016, pp. 676-679.

[13] Kazutoshi Wakabayashi “Real-Time with C Com- piler for Hardware.”

[14] Zhaoqing Wang and H. H. Cheng, "Portable C/C++ code for portable XML data," in *IEEE Software*, vol. 23, no. 1, pp. 76-81, Jan.-Feb. 2006.

[15] K. G. Cheetancheri and H. H. Cheng, "Mixed Language Programming in C/C++ and Java for Applications in Mechatronic Systems," *2006 2nd IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications*, Beijing, 2006, pp. 1-6.