



Maratha Vidya Prasarak Samaj's
Karmaveer Adv. Baburao Ganapatrao Thakare
College Of Engineering
Nashik-13.

(NAAC ACCREDITED INSTITUTE WITH 'A' GRADE)



DEPARTMENT OF ELECTRONICS & TELECOMMUNICATION ENGG.

Departmental TeChronicle

Month: - September 2021

Vol. - 03, Issue – 3

Department Vision:-

To be recognized as an excellent department offering competent technical education to create competent electronics & telecommunication engineers for the benefit of the common masses.

Department Mission:-

Committed to serve the needs of society through innovative teaching learning processes, promoting industry-institute interaction to provide competent and cultured electronics and telecommunication engineers.

Program Educational Objectives:-

1. To impart state of art technical education in the Electronics & Telecommunication Engineering.
2. To promote society beneficial projects and activities.
3. To develop soft skill, team work, professional ethics and multidisciplinary approach for the carrier enhancement.
4. To bridge the gap between Industry-Institute through collaboration with Industries, Institutions and Universities.
5. To provide suitable infrastructure and facilities in tuned with advancing technological evaluation.

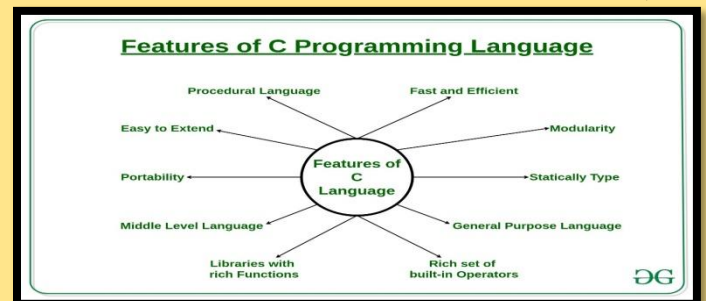
Greeting,

Department of Electronics and Telecommunication Engineering is celebrating Engineers Day by unveiling technical newsletter "TeChronicle" VOL3, ISSUE-3 on 15th September 2021. Every year this day is celebrated as to appreciate the exceptional contributions of Sir Mokshagundam Visvesvaraya.

EVERGREEN LANGUAGE: C LANGUAGE

[Prachi Jorwar TE E&TC]

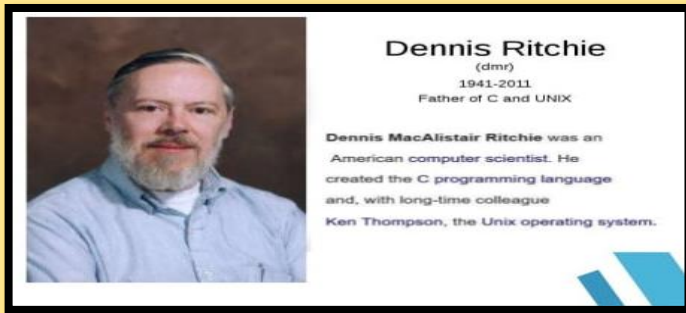
C is a general-purpose programming language that is used for creating a variety of applications. C is the pillar of the programming language. Earlier C language had the reputation of being the language of OS, but later on few programmers tried to write some general programs, and they came to the conclusion that the language can be used for general programming without any problems, and they started using it.



C language is often called as a middle level language because it includes features of both low level or machine languages and high-level programmer friendly languages. This language has features like it is a fast, flexible and structured programming language having a rich library. Because of these features, C is also considered as a general-purpose programming language, and therefore has a vast application area in various domains.

Although it seems that the position of C language has been affected because of the growing popularity

of programming languages like Java and C++, however C has managed to cement its position in the programming world, and it is going to last longer than any other language. The major reason behind this is that all the syntaxes and features of C language have been inherited by C++. The only difference between C and C++ is that C++ has implemented an advanced programming paradigm that is popularly known as Object Oriented Programming. Therefore, if you are interested in learning C++, you have to first learn C language.



Some people have the view that due to the procedure-oriented nature of C language, one has to leave the procedure programming habits if he/she wants to learn object-oriented programming. In addition, the popularity of the GUI based C++ programming environments is more because of their simplicity. But there is a drawback associated with that, these environments consume a large amount of memory, resulting in reduced efficiency.

Moreover, the smaller programs that execute on smaller systems are made using complex libraries, which is not a suitable way. So, when it comes to writing device driver software or other utility applications, or embedded applications, the preferred choice is C language instead of C++. So, even if the new programming languages have affected the position of C language, it still remains in the race and will continue for a longer period of time.

The origin of C is closely tied to the development of the Unix operating system, originally implemented in assembly language on a PDP-7 by Dennis Ritchie and Ken Thompson, incorporating several ideas from colleagues. Eventually, they decided to port the operating system to a PDP-11. The original PDP-11 version of Unix was also developed in assembly language.[6]

Thompson desired a programming language to make utilities for the new platform. At first, he tried to make a Fortran compiler, but soon gave up the idea. Instead, he created a cut-down version of the recently developed BCPL systems programming language. The official description of BCPL was not available at the time,[11] and Thompson modified the syntax to

be less wordy, producing the similar but somewhat simpler B. [6] However, few utilities were ultimately written in B because it was too slow, and B could not take advantage of PDP-11 features such as byte addressability.

In 1972, Ritchie started to improve B, which resulted in creating a new language C. [12] The C compiler and some utilities made with it were included in Version 2 Unix.[13] At Version 4 Unix, released in November 1973, the Unix kernel was extensively re-implemented in C. [6] By this time, the C language had acquired some powerful features such as struct types.

Preprocessor was introduced around 1973 at the urging of Alan Snyder and also in recognition of the usefulness of the file-inclusion mechanisms available in BCPL and PL/I. Its original version provided only included files and simple string replacements: #include and #define of parameterless macros. Soon after that, it was extended, mostly by Mike Lesk and then by John Reiser, to incorporate macros with arguments and conditional compilation.[6]

Unix was one of the first operating system kernels implemented in a language other than assembly. Earlier instances include the Multics system (which was written in PL/I) and Master Control Program (MCP) for the Burroughs B5000 (which was written in ALGOL) in 1961. In around 1977, Ritchie and Stephen C. Johnson made further changes to the language to facilitate portability of the Unix operating system. Johnson's Portable C Compiler served as the basis for several implementations of C on new platforms.

References:

- <https://www.geeksforgeeks.org/c-programming-language/>

Extended Version of C language: C++
[Tanvi Patil TE E&TC]



C is the mother of all the programming language. C is the general purpose programming language

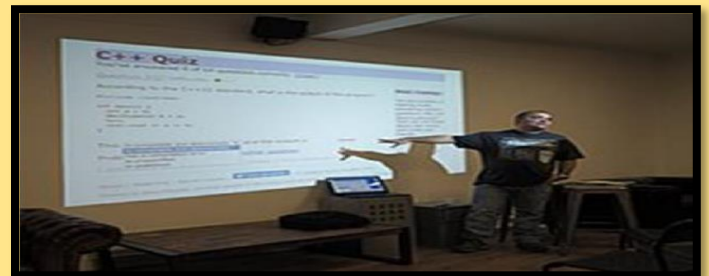
which is generally used for creating variety of application. This language was originally developed for writing operating systems. Basically, C++ is the extended version of C. C++ has an advanced programming paradigm that is popularly known as Object Oriented Programming. C++ is the high-level computer programming language, which is developed by Bjarne Stroustrup of Bell Laboratories in the early 1980's, it is based on the traditional C language but with added object –oriented programming and other capabilities. C++, along with Java, has become Popular, developing commercial software packages incorporate multiple interrelated applications.



Bjarne Stroustrup, the creator of C++, in his AT&T New Jersey office c. 2000

In year 1979, Bjarne Stroustrup a Danish computer scientist started working on 'C' with Classes, the forerunner to C++. Stroustrup found that Simula (Simula – Complete History of Simula Programming Language Simula The first object-oriented programming language was developed in the 1960s at the Norwegian) has helpful features for large software development, but language was very slow for practical use, while BCPL (BCPL is a procedural, imperative, and structured programming language. Originally intended for writing compilers for other languages, BCPL is no longer in common use) was fast but too low-level to be suitable for large software development. When Stroustrup started working in AT&T Bell Labs he started facing the problem of analysing the UNIX kernel with respect to distributed computing, so after remembering his PhD experience, Stroustrup hit the road to inflate the C language with Simula – like idiosyncrasy. C was chosen as it is portable language, it has quality to extend itself, Open-source and widely used. As well as C and Simula's influences, other languages also influenced this new language, including ALGOL 68, Ada, CLU and ML.

In 1982, Stroustrup started to develop a successor to C with Classes, which he named "C++"; (++ being the increment operator in C) after going through several other names. In 1985, the first edition of The C++ Programming Language was released, which became the definitive reference for the language, as there was not yet an official standard. In 1989, C++ 2.0 was released, followed by the updated second edition of The C++ Programming Language in 1991, New features in 2.0 included multiple inheritance, abstract classes, static member functions, const member functions, and protected members. On January 3, 2018, Stroustrup was announced as the 2018 winner of the Charles Stark Draper Prize for Engineering; for conceptualizing and developing the C++ programming language as of 2021 C++ ranked fourth on the TIOBE index, a measure of the popularity of programming languages, after C, Java, and Python. C++ has the same advantages as C, but with more features. Programs that require speed, scalability, and are not massive, should consider using C++. We should have a good idea of when it might be appropriate to use each language.



A quiz on C++11 features being given in Paris in 2015

Reference:

- <https://www.geeksforgeeks.org/c-plus-plus/>

R (Programming Language)

[Yash Yeola TE E&TC]

R is a programming language and free software environment for statistical computing and graphics for Statistical Computing. It is widely used among data miners and statisticians for developing data analysis and statistical software. Polls, data mining surveys, and studies of scholarly literature databases show substantial increases in R's

popularity; since August 2021, R ranks 14th in the TIOBE index, a measure of popularity of programming languages. The official R software environment is a GNU package.



R and its libraries implement various statistical and graphical techniques, including linear and nonlinear modelling, classical statistical tests, spatial and time-series analysis, classification, clustering, and others. R is easily extensible through functions and extensions, and its community is noted for its active package contribution. Many of R's standard functions are written in R itself, which makes it easy for users to follow the algorithmic choices made. For computationally intensive tasks, C, C++, and Fortran code can be linked and called at run time. Advanced users can write C, C++, Java, .NET or Python code to manipulate R objects directly. R is highly extensible through the use of user-submitted packages for specific functions and specific areas of study. Due to its S heritage, R has stronger Object-oriented facilities than most statistical computing languages. Extending it is also facilitated by its lexical scoping rules.

Another of R's strengths is static graphics; it can produce publication-quality graphs, including mathematical symbols. Dynamic and interactive graphics are available through additional packages.

The R Environment

R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes

- an effective data handling and storage facility,
- a suite of operators for calculations on arrays, in particular matrices,

- a large, coherent, integrated collection of intermediate tools for data analysis,
- graphical facilities for data analysis and display either on-screen or on hardcopy, and
- a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.

Open-source, Strong graphical capabilities, highly active community, Wide selection of packages, Comprehensive environment, perform complex statistical calculations, Distributed computing, interfacing with databases, Cross-platform support, Compatible with other programming languages, Data handling and storage, Compatibility with data processing technologies, Generates report in any desired format.

Reference:

- <https://techvidvan.com/tutorials/why-r/>
- <https://www.r-project.org/about.html>

Swift Programming Language:

[Dhirajkumar Saindane TE E&TC]

This Particular article is about Latest Programming Language, which we have to Essential in Factual World Scenario.

“One Step Towards The Technology Globe” Swift Programming Language has been Approved/Developed by the Apple IOS Inc. and it's Open Community Source in 2014. In Recent Times of this Language, Especially Used for IOS and OSX Application Development. First Version of Swift is 1.0, On Now Currently Accepted Version is 5.4.2. Swift Comes to make ios and osx (macOS, iPadOS ,watchOS , tvOS) Development Simpler and more Fun.

Swift Programming Language is being useful for Top Organizations Apps Built. Here Are the Top Concerns/Applications using Swift:



Facebook: Social Collaborative Platform.

Uber: Multinational ride-hailing company.

WhatsApp: Swift is One of the Prime Language to contributed that has secured and most popular.

LinkedIn: LinkedIn is depended on Swift for the Robust platform. Instagram: It is useful for photos and sharing album, it leverages Swift.

Twitter: The agile nature of Swift is leveraged in this iOS-based Twitter application.

Kickstarter: It uses Swift to recognise latest projects in areas of various interests.

True caller: It is Known to be using Swift in its Fabrication, True caller for Apple devices can recognise or block spam calls and search for unknown numbers.

VSCO: An Amazing photo and video editor, VSCO uses Swift for its Formation.

Such as above , Many application it is valuable on its Own using Swift programming Language: Airbnb , Bitmoji , Dubsmash ,CNN, Fitbit Medium , Pandora , On A Closing Note , etc . As we all know Apple is Popular and Leading Company. it is most Secured Compare to all Inc. Apple already had a programming Language i.e., Objective-c. So why apple introduce new (Swift) Programming Language? Well, When Objective-c was created some Functions might not be clean, well-shaped so it's less Communicate to the Programmers so that's why it does not convey today's taste for a Language. Objective-C is lacking some of these things like a clean syntax (and Syntactic sugar) and type inference. Swift tries to fill this gap.

Features: Swift is Object Oriented Programming language and it has short Syntax so it must be useful for freshers Programmers, they can easily Access it. Apart from that we want to learn that language, we have gone through IDE (Integrated development

Environment). This IDE named as X-Code. IDE helps to Providing Various Tools to the Developers so they can use to get fast process Application.

Advantages of Swift compare to Objective C:

1] In objective-c Syntax, we have to declared an instance variable but swift doesn't need to declared instance variable.

2] In Objective-C, Execution time is taking more and it is compiling code by code But in Swift execution and compiling time is fast What's mean that After the Compilation of code we just edit some code and In Swift language preferred to only compiling the editing code not a whole program code step by step that is the major and profitable advantage for the factual world. 3] In Objective-c, we have to declare the variable datatype (Integer/float/String) in a program but as compared to Swift, we only have to do gives input variable to the program, this system automatically detect this variable is integer or other.

Reference:

- <https://www.infoq.com/articles/apple-swift/>
- <https://www.spec-india.com/blog/top-appsbuiltusing-swift>

Committee Members	
Dr. Vijay M. Birari	Editor in Chief
Ms. T. S. Deshmukh	Co-Editor
Mr. Viraj R. Sonawane	Staff Coordinator
Ms. Prachi Jorwar	Student Co-Editor
Mr. Yash Yeola	Student Coordinator
Ms. Tanvi Patil	Student Coordinator
Mr.Dhirajkumar Saindane	Student Coordinator

Website: www.kbtcoe.org

Email Id: techronicle.etc@gmail.com