International Journal Of Computer Architecture And Mobility (0065-0131)

Volume 1-Issue 1, November 2012

Comparative study of languages

Swati Joshi , *SDBCE (Indore)*

swatijoshi6688@yahoo.co.in

***Abstract*—A programming language is an artificial language designed to communicate instructions to a machine, ,particularly a computer. Programming languages can be used to create programs that control the behavior of a machine and/or to express algorithms precisely. A programming language includes its Functions and targets, Abstraction and its expressive power. There are two categories according to their interpretation it is Low level Language and High Level language, which creates base for language creation . This paper deals with the comparative study of languages like Pascal , Fortran , Cobol, Python, java , C ,C++. Different languages have there own compilers and there own properties . Pascal is a structured and syntactically strict language, also it is data oriented. Fortran is a general purpose language used for mathematical computation, also portable across machine platforms. Python is high level, interpreted, interactive and object oriented scripting language. COBOL is designed for developing business, typically file-oriented, applications. C is a procedural programming language.** C++ **is a statically typed, free-form, multi-paradigm, compiled, general-purpose programming language, also includes object oriented features into it. Java is a general-purpose, concurrent, class- based, object-oriented language .This paper provides comparison of various languages that can help various researchers to deal with them.**

***Keywords*— PYTHON, FORTRAN, PASCAL, COBOL, C , C++, JAVA**

I. Introduction

Computer languages are a subset of programming language. These are designed for a specific class of computers. These languages have their own compilers and properties. They are categorized in high level language and low level language. **[1]** Earlier in late 1940s first electronic computers & LLLs are developed, then in 1950s first HLLs for computers then in 1969 about 120 HLLs and, about 15 in widespread use then 1977about 80 HLLs in active (non-trivial) use and finally Today more than 2000 HLLs are developed. The most important differences between various languages are syntax and semantics.

Syntactic differences are saying the same thing in different ways For Example:-

```
Infix
        a = b + c
Prefix
        (set a, (add b, c))
Postfix
        b c + a =
"English"
        ADD b TO c GIVING a
Distributed
         PAR
            to_a ! b + c to_a ?
            a
```

Semantic difference are saying different things in different paradigms as: Imperative paradigm, object oriented paradigm, functional paradigm, Logic paradigm , parallel paradigm, scripting paradigm.

Imperative paradigm: The basic ideas are about describing state, the actions that modify the state, and the sequence of events.

Object oriented Paradigm: This is similar to Imperative but with maximum use of types & scopes - keep state in objects, each type of object having its own set of methods. Furthermore, the state in an object can often only be altered via its associated methods.

Functional Paradigm: Functional languages emphasise the transformations of values.

Logic Paradigm: Define a problem by describing its facts and properties, and then solve it by giving the system a goal to prove using those facts and properties.

Parallel Paradigm: In parallel languages, if actions don't interact, work on them in any order(can be non-deterministic), or even simultaneously (multi-processing).

Scripting Paradigm: It includes situations as - Building applications by clubbing existing components, To control applications, To develop program with ease of development.

II. Features of Languages

A Language is characterised by its features. The features of the Languages are as follows:

The Pascal programming language was originally developed by Niklaus Wirth.**[2]** His principle objectives for Pascal were for the language to be efficient to implement and run, allow for the development of well structured and well organized programs, and to serve as a vehicle for the teaching of the important concepts of computer programming.

Pascal contains some significant language features - (that allow it to used as a powerful learning tool in introducing structured programming techniques) Built in Data Types, User defined Data Types , Provides a defined set of Data Structures, Has a strong data typing element , Supports Structured Programming , Simplicity and Expressivity**.**

Fortran is the oldest programming language, developed by a team of programmers at IBM led by John Backus, and was first published in 1957. **[3]** The name FORTRAN is an acronym for FORmula TRANslation, because it was designed to allow easy translation of math formulas into code

Fortran includes following significant features - Simple to learn, Machine Independent, More natural ways to express mathematical functions ,Problem orientated language, Remains close to and exploits the available hardware, Efficient execution, Ability to control storage allocation, More freedom in code layout.

.

Python[4] is an interpreted, interactive, object-oriented programming language. It incorporates modules, exceptions, dynamic typing, very high level dynamic data types, and classes. Python combines remarkable power with very clear syntax. It has interfaces to many system calls and libraries, as well as to various window systems, and is extensible in C or C++. It is also usable as en extension language for applications that need programming interfaces. Finally, Python is portable across all major hardware and software platforms.

[11]COBOL (Common Business Oriented Language) was one of the earliest high-level programming languages. It includes - The language that automated business, Allows names to be truly connotative, Every variable is defined in detail , File records are also described with great detail, as are lines to be output to a printer, Offers object, visual programming environments, Class Libraries, Rapid Application Capabilities, Integration with the World Wide Web.

C [5] is a powerful, flexible language that provides fast program execution and imposes few constraints on the programmer. It allows language for both systems programming and general purpose programs. C's power and fast program execution come from it's ability to access low level access to information and commands while still retaining the portability and syntax of a high level language. These qualities make it a useful low level commands, similar to assembly language, but with high level syntax. It's flexibility comes from the many ways the programmer has to accomplish the same tasks. C includes bitwise operators along with powerful pointer manipulation capabilities. C imposes few constraints on the programmer. The main area this shows up is in C's lack of type checking. This can be a powerful advantage to an experienced programmer but a dangerous disadvantage to a novice. Another strong point of C is it's use of modularity. Sections of code can be stored in libraries for re-use in future programs. This concept of modularity also helps with C's portability and execution speed. The core C language leaves out many features included in the core of other languages. These functions are instead stored in the C Standard Library where they can be called on when needed.. An example of this concept would be C's lack of built in I/O capabilities. I/O functions tend to slow down program execution and also be machine independent when running optimally. For these reasons, they are stored in a library separately from the C language and only included when necessary.

C++ [6] programming languages is an extension of C that was developed by Bjarne Stroustrup in the early 1980s at Bell Laboratories. C++ provides a number of features that smarten up the C language, but more importantly, it provides capabilities for object-oriented programming. Object-oriented programs are easier to understand, correct and modify. Many other object-oriented languages have been developed, including most notably, Smalltalk. The best features of C++ are - C++ is a hybrid language-it is possible to program in either a C-like style, an object-oriented style, or both., C++ programs consist of pieces called classes and functions. You can program each piece you may need to form a C++ program. The advantage of creating your own functions and classes is that you will know exactly how they work. You will be able to examine the C++ code.

[12] Java programming Language evolved from a language named Oak. Oak was developed in the early nineties at Sun Microsystems as a platform-independent language aimed at allowing entertainment appliances such as video game consoles and VCRs to communicate. It`s features are - Platform Independence, Object Orientation, Rich Standard Library (*Language Support* classes for advanced language features such as strings, arrays, threads, and exception handling, *Utility* classes like a random number generator, date and time functions, and container classes. *Input/output* classes to read and write data of many types to and from a variety of sources. *Networking* classes to allow inter-computer communications over a local network or the Internet, *Abstract Window Toolkit* for creating platform-independent GUI applications, *Applet* is a class that lets you create Java programs that can be downloaded and run on a client browser), Applet Interface, Familiar C++-like Syntax, Garbage Collection.

III. Problems of languages

Problems of the languages are the deficiency or some of the pitfalls of the language.

Pascal includes some of the major problems - It seems the array sizes and string lengths were part of the type so it was not possible to create functions that would accept variable length arrays or even strings as parameters, Booleans expressions were wrongly interpreted, Library was not sufficient as it lacks static variables and other smaller contexts.

Fortran MPI provides operations that may be hidden from the user code and run concurrently with it, accessing same memory as user code.

Python includes the main problem is its memory management model, it uses refrence counting method and there are many drawbacks using this method if refrence to any pointer is lost the whole memoryattached with that pointer is deallocated ,while other languages uses memory leaks by circular refrence . COBOL did not completely cover complex mathematical calculations, and manipulation of single bits of information. It also lacks in Graphical Screen Design, Lacks affordable Integration in Bussiness Application.

C language has the problems like its not object oriented, no string datatype is defined, it includes functions for insignificant operations, Overflow of Integer type variable without warning, It is inconsistent with error handling, Relies more on Preprocessor power.

C++ language also have problems like lacking in pointing for member functions, it has no run time and compile time encapsulation, it has defective operator overloading, Garbage collection is manual creating clumsy programming, Libraries are not self sufficient, it also lack in standardisation of Binary levels.

Java problems includes native unsigned integer types which prevents direct interchange of data between C and Java platform. It lacks the property of structure of C/C++, Large array size results in compile- time errors, it lacks the property of isolation in general i.e. not providing user isolating from server programs.

IV. Applications Areas Of Languages

PASCAL:- The Prime area of application that Pascal entails is the learning environment. Now with the growing popularity of Object Orient Programming Pascal has taken a back seat to other languages such as C++ and Visual Basic.

FORTRAN:- FORTRAN is useful for a wide variety of applications, some of the more outstanding ones are as follows :- Number crunching (due to the more natural (like it's true algebraic form) way of expressing complex mathematical functions and it's quick execution time, FORTRAN is easy and efficient at processing mathematical equations). Scientific, mathematical, statistical, and engineering type procedures (due to it's rapid number-crunching ability FORTRAN is a good choice for these type of applications) Basically FORTRAN is most useful for applications that are *"computational-bound"* rather than *"I/O bound"*.

PYTHON:- Python is ideally suited for rapid prototyping of complex applications. It is also used as a "glue language" for connecting up the obvious pieces of a complex solution, such as Web pages, databases, and Internet sockets.

COBOL:- COBOL is ideally suited for the solution of business problems. It is interesting to note that COBOL was the first programming language whose use was mandated by the Department of Defense (DoD).

C:- The C programming language is used in many different areas of application, but the most prolific area is UNIX operating system applications. The C language is also used in computer games.

C++:-C++ provides a collection of predefined classes, along with the capability of user-defined classes. The classes of C++ are data types,which can be instantiated any number of times. Class definitions specify data objects (called data members) and functions (called member function). Classes can name one or more parent classes, providing inheritance and multiple inheritance, respectively. Classes inherit the data members and member functions of the parent class that are specified to be inheritable. Therefore it is mainly used for Software Engineering, Graphics.

JAVA:- Java has varios areas of applications. Some areWorld Wide Web Applets, Cross-Platform Application Development, Other Network Applications.

V. Advancement And Future Aspects Of Languages

[8]In 1960s Algol language is developed which is platform independent and used for scientific computing.. Algol is extended to pascal in 1970s whichis becoming object oriented. Pascal is data oriented, giving the programmer the abiity to define custom data types. With this freedom comes strict type checking, which prevented data type from being mixed up. Pascal was intended as a teaching languageand adopted also. From version 1.0 to 7.0 of Turbo Pascal , Borland continued to expand the language. Then in 1970s C is developed for unix operating system by AT&T Bell labs. Pascal was finally killed by object orientation . In 1980s C++ came in to picture to define object orientation to family of programmers. [9]Fortran was first ever high level programming language . Work on fortran started in 1950s , Fortran version is denoted by the last two digits of the year it is developed.Most common version still in use is FORTRAN 77, although FORTRAN 90 is growing its popularity. Python is high level ,Interpreted , interactive, and object oriented scripting language. [10]Its latest versions are python 2.7.3 and python 3.3.0, they both are the most stable releases. [11]COBOL developed in 1959Since 1959 it has undergone several modifications and improvements. In an attempt to overcome the problem of incompatibility between different versions of COBOL, the American National Standards Institute (ANSI) developed a standard form of the language in 1968. This version was known as American National Standard (ANS) COBOL. In 1974, ANSI published a revised version of (ANS) COBOL, containing a number of features that were not in the 1968 version. In 1985, ANSI published still another revised version that had new features not in the 1974 standard. The language continues to evolve today. Object-oriented COBOL is a subset of COBOL 97, which is the fourth edition in the continuing evolution of ANSI/ISO standard COBOL. COBOL 97 includes conventional improvements as well as object-oriented features. Like the C++ programming language, object-oriented COBOL compilers are available even as the language moves toward standardization.

VII. Conclusion

The Comparative study of languages deals with the Languages Python, Fortran, Pascal, Cobol, C, C++, Java. It describes features, problems, advancement, and application of these Languages which helps other reasearchers to deal and elaborate these Languages.

VIII. References

1. The computer language [online] : Available “ http://www.cs.man.ac.uk/~pjj/cs1001/software/node3.html”

2. The pascal tutorial [online]: Available “ http://groups.engin.umd.umich.edu/CIS/course.des/cis400/pascal/pascal.html”

3. The Fortran tutorial [online]: Available “ http://groups.engin.umd.umich.edu/CIS/course.des/cis400/fortran/fortran.html”

4. The python tutorial [online]: Available “ http://groups.engin.umd.umich.edu/CIS/course.des/cis400/python/python.html”

5. The C tutorial [online]: Available “ http://groups.engin.umd.umich.edu/CIS/course.des/cis400/c/c.ht ml”

6. The C++ tutorial [online]: Available “ http://groups.engin.umd.umich.edu/CIS/course.des/cis400/cpp/cp p.html”

VI. Result

	Pascal	Fortran	Python	C	C++	Java	Cobol
Platform independence	4	5	4	5	5	5	1
Efficiency	3	4	2	5	4	5	3
Usefullness	3	2	3	3	4	4	4
I /O Facilities	4	4	2	3	4	4	2
Simplicity	4	4	2	3	4	4	5

1- Poor
2- Weak
3- Good
4- Very Good
5- Excellent

7. The Cobol tutorial [online]: Available “ http://groups.engin.umd.umich.edu/CIS/course.des/cis400/cobol/cobol.html”

8. Gbadeyan J.A , Sadiku J.S , AND Babatunde A.U “Analysis of Pascal, Basic AND programming Language

“http://www.unilorin.edu.ng/publications/babatundeao/ANALYSIS%20OF%20PASCAL.pdf”

9. The Pascal Tutorial by TAOYUE in 2011 http://www.taoyue.com/tutorials/pascal/history.html

10. T.F Coleman And C. Van Loan , “Handbook for matrix computation” Published by SIAM “http://www.geo.utexas.edu/courses/387h/Lectures/fortran%2077%20tutorial.pdf”

11. The Python community http://www.python.org/getit/

12. The java tutorial [online]: Available http://groups.engin.umd.umich.edu/CIS/course.des/cis400/java/java.html