

# ONE ALGORITHM TO MANY PROGRAMMING LANGUAGE

Manumon Mathew<sup>1</sup>, Ms. Gloriya Mathew<sup>2</sup>

<sup>1</sup>PG Scholar of Amal Jyothi College of Engineering, Kanjirappally, Kerala

<sup>2</sup>Asst. Professor of Amal Jyothi College of Engineering, Kanjirappally, Kerala

**Abstract:** *As with the advancement in technology creates new languages. Learning all the languages is not impossible but hard to achieve. This paper depicts an algorithm which generates the source code of a particular programming language as specified with respect to the algorithm of a predefined format. This can help many programmers to work with different languages without knowing the actual syntax of that language. This system is flexible to use and contains only simple steps to convert the algorithm to a programming language.*

**Keywords:** Common-Syntax, Inter-conversion, Pattern-Matching, Syntax Parser

## I. INTRODUCTION

In this fast growing world, many languages are equally important. Most of today's IT Companies work in different languages. So each time the employees have to learn new languages. There are some software's that are capable of converting one language into another language. But they are at a disadvantage of converting to an only a specific language. So if there is a language that can be converted to many language, then it can save lot of time and can be very useful in most of the IT industries today.

This paper proposes a method to convert one simple algorithm like language to any other programming language. The input algorithm is a simple language where the syntax is similar to the usual algorithm. It contains a debugger of it's on. So there will not be any syntax error in the converted programming language. We can also save our output programming language. With this system, learning syntax of new language is not

necessary. All that the system require is a simple algorithm with logic and the system will convert the algorithm to any language syntax as the user prefer.

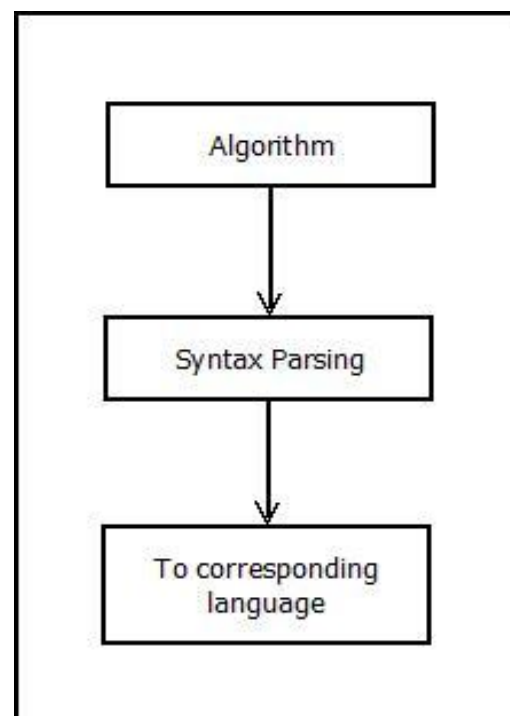


Fig -1: Basic converting Process

The basic idea (Fig-1) behind the system is to take the algorithm as its input and pass the algorithm to the syntax parser for a syntax error checking. If no error is found then it converts the algorithm to the corresponding language.

## II.LITERATURE REVIEW

### **Language Modernization from C to JAVA[1]:-**

Jadhav et al, proposed a system to convert C language to Java. This was produced for organizations who are utilizing mainframe systems. This mainframe system utilizes languages like C. So as to change over these languages to progressively adaptable language like Java, the conversion method can be utilized. Java has got many features than other mainframe systems like languages.

Changing over from C to Java is impossible straightforwardly since C is procedural and Java is object-oriented. So first they changed over C to C++, which is an item arranged and the sentence structure of C and C++ is comparatively similar, since they originate from a similar family. After converting C to C++, this C++ is changed over the Java program and it will be to some degree simpler as highlights of both C++ and JAVA like classes, objects, legacy, and so forth are comparable. Consequently, we can get the C code changed over into JAVA.

### **Programming Language Inter-conversion[2]:-**

George et al, developed an idea that programming language can be changed between wide ranges of other programming language. The thought is about the execution of the middle of the road language for between changes. This language can be utilized to store the logic of the program in an algorithmic arrangement without aggravating the structure of the base program.

The transformation procedure is an iterative procedure, i.e., like a compiler, there are a few procedures engaged with this. For a superior transformation process, the framework ought not to be only a basic grammar substitution framework. It ought to likewise protect the structure or ought to change it to make it far better by expelling repetitive codes.

**Compiler Design[3]:** -Chhabra et al, idea of compiler design was compiler transforms as well as orders a program written in a reasonable source language into a proportional target language through various stages. Beginning with the recognition of token through target code give a

premise to a correspondence interface between a client and a processor immaterial measure of time. Comprehension of these connections facilitates the unavoidable advances to new equipment and programming language and improves an individual's capacity to make a proper exchange off in structure and usage. A large number of the methods used to build a compiler are valuable in a wide assortment of utilizations including representative information.

**Syntax parsing[4]:-** Shuet et al, proposed idea was syntactic parsing manages the syntactic structure of a sentence. The word 'linguistic structure' suggests to the syntactic course of action of words in a sentence and their association with one another. The goal of the syntactic investigation is to locate the syntactic structure of a sentence which is generally portrayed as a tree. Distinguishing the syntactic structure is valuable in deciding the significance of a sentence. Regular language preparing is a field of software engineering and semantics, worried about the dealings among PCs and human dialects. It forms the information through lexical investigation, Syntax examination, Semantic investigation, Discourse preparing, Pragmatic examination.

## III.OBJECTIVE OF THE PROJECT

This project is to help programmers to program with multiple languages. i.e., the program has to know the logic and to implement that logic in a simple algorithm. Then the algorithm can be converted to any programming language using the proposed system. This system converts the logical algorithm to a well-structured programming language.

The application for this system is numerous. There are companies that use legacy programming to make their software's. So in order to use a new programming language, they have to hire new employees. But with this proposed system, they can easily implement the new programming language in their software. This system creates a common-syntax for all programming language.

## IV.METHODOLOGY

The proposed system input algorithm is in simple and basic algorithm format. The system basically

uses pattern matching for parsing technique and algorithm conversion. So it may take some time depending on the algorithm's total number of lines.

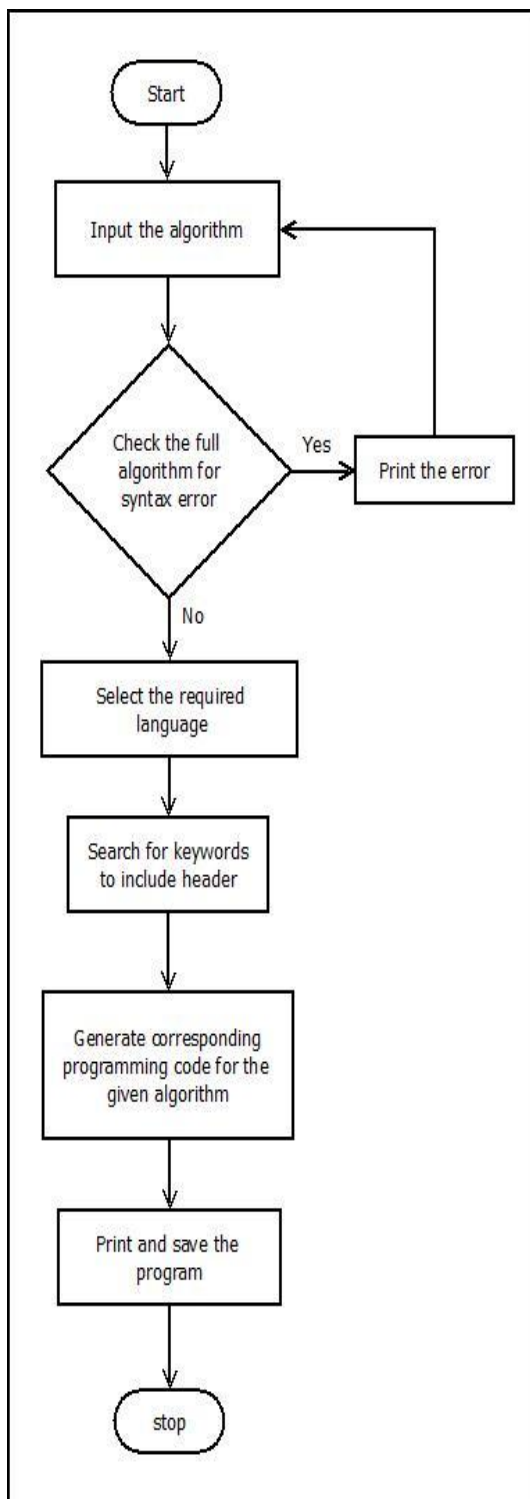


Fig -2: Flow Chart of the converting system

The flow of the system (Fig-2) is as follows:

First, the system takes a simple algorithm as input. Then the system checks for all syntax errors in full algorithm. The algorithm format is predefined in

the system. If the input algorithm doesn't match with the predefined algorithm the system returns all the errors with the line number. Then the user can correct this error and resubmit this algorithm.

After syntax error checking for the input, the user can select the programming language for conversion. The system also provides a facility for saving the converted language in a file. The user have to enter the file name and the corresponding extension will be automatically added to the converted program.

Then the system checks full algorithms for keywords that require the corresponding header to be included in the program. With this, the system includes all the headers like the math header for using math's function in the program.

Then the system starts to compare each line of the algorithm, with the corresponding programming algorithm saved in the system for conversion. When comparing matches the corresponding programming code is displayed and saved in a file. This continues until the end of the algorithm.

Finally the converted program will be displayed on the output monitor. The output program is saved in a file and the user can use that file for future works.

## V.IMPLEMENTATION

The proposed system is implemented in CPP. It is a general-purpose programming language and widely used nowadays for competitive programming. It has imperative, object-oriented and generic programming features.

The proposed system implementation is currently a prototype that is capable of converting algorithm to 3 languages including C, CPP, and Python with basic features like loops, if statement, math function, user-defined function.

For example an algorithm for adding 2 numbers:

```

start
declare a,b,c as integer
print 'Enter 2 Numbers:'
get a,b
let c=a+b
print 'sum is:'
print c
  
```

stop

This will be converted to C++ in:

```
#include<iostream>
using namespace std;
main()
{
    int a,b,c;
    cout<<"Enter 2 Numbers:";
    cin>>a>>b;
    c=a+b;
    cout<<"sum is:";
    cout<<c;
}
```

This will be converted to Python in:

```
print('Enter 2 Numbers:')
a=eval(input())
b=eval(input())
c=a+b
print('sum is:')
print(c)
```

Similarly, it can be converted to C language too.

## VI.CONCLUSION

Nowadays almost in all sectors, computer programming plays a big role. Many programming languages are developed in each year. Some are secure, some are more powerful than the older ones. So it essential to be updated and the proposed system helps us to do just that. i.e., it helps the developer to develop software's in different programming languages without knowing its syntax. This system can be also used as study material for learning new languages. So with this system, developers with logic can easily program in any language.

## VII.FUTURE SCOPE

The focus of future work is to improve the existing system and finding the proficient way for converting between the programming languages itself. i.e., converting one programming language to any other programming language.

The development of application in mobile phones for this proposed system can be used to convert between programming languages for easy interface.

## REFERENCES

- [1] Pragati Y. Jadhav, Shital Jadhav, Sonal Derle, Sonali Jawarestudent of GESRHSCOE, Nashik of the Savitribai Phule Pune University, "Language Modernization from C to JAVA", International Research Journal of Engineering and Technology (IRJET), Volume: 05 Issue: 01 | Jan-2018
- [2] Dony George, Priyanka Girase, Mahesh Gupta, Prachi Gupta, Aakanksha Sharma students of FCRIT, Vashi, Navi Mumbai "Programming Language Inter-conversion", International Journal of Computer Applications (0975 – 8887) Volume 1 – No. 20.
- [3] JatinChhabra, Hiteshi Chopra, Abhimanyu Vats Research paper on "Compiler Design" 2014 IJIRT | Volume 1 Issue 5| ISSN: 2349-6002
- [4] Xinhua Shu, Madhuri A. Tayal, Dr. M. M. Raghuvanshi, Dr. Latesh Malik "Syntax Parsing: Implementation using Grammar-Rules for English Language" published on 2014 in International Conference on Electronic Systems, Signal Processing and Computing Technologie
- [6] Programming language available at: [http://en.wikipedia.org/wiki/Programming\\_language](http://en.wikipedia.org/wiki/Programming_language)
- [7] CPP format at <https://www.geeksforgeeks.org/c-plus-plus/>