North South University

Department of Electrical & Computer Engineering

Project Report

Course Code: CSE 425

Course Title: Concepts of Programming Language

Section: 01

Project Title:

Bangla Programming Language, designed for Students of Secondary Education Level.

Date of Submission: December 24, 2024

Submitted by:

01. Joy Kumar Ghosh – 2211424 6 42
02. Ahanaf Tahomid – 1831920 6 42

Submitted to:

Course Instructor, Dr. Md Shahriar Karim (MSK1)

Abstract:

In the current world, learning a programming language has become crucial for almost every student from different disciplines. In Bangladesh, students start learning programming languages at the intermediate level and face difficulty understanding the fundamentals of a programming language. Suppose we can push them to start learning a programming language in their native language 'Bangla' at the secondary level. In that case, it will help them to understand the advanced language at the intermediate level as well as at the university level. This project aims to develop a "Bangla Programming Language" based on a compiler by keeping the difficulty level as minimal as possible. So that students can transition to advanced programming languages like C, C++, Java, and Python more smoothly. In this primitive stage, we assembled some fundamental operations, including assignments, conditionals, loops, arrays, and some arithmetic operations using the complete Bangla Language.

Furthermore, we tried to keep it more orthogonal so that users can define identifiers and numbers using both Bangla and English literals but so far, we only allowed Bangla Literals for source code. Many programmers in India and Bangladesh have built the Bangla Programming Language, but most languages are built based on interpreters like Java. We tried to build a compiler based on cpp. Our Bangla Programing Language is currently working as an interpreter based on cpp.

Introduction and Literature Review:

Learning a programming language is crucial for every student; the learning resources are primarily available in English, which creates a barrier for students in non-English-speaking countries. In Bangladesh, many intermediate-level students struggle to learn programming languages because most languages use English keywords and documentation. In Bangladesh, more than 90% of students study in their native language, "Bangla." English is part of the curriculum from standard III (third grade) up to Class XII, and they must learn English in the university. This language transition initially becomes a significant barrier for young learners and often demoralizes many bright students. Furthermore, many students start to explore programming language for the first time, and the language barrier gives them a slower starting point.[02] This project aims to create a programming language that uses Bengali keywords and syntax, making it easier for Bengali speaking students to grasp fundamental programming concepts.

Various nations have attempted to introduce their local languages as programming languages with an educational focus. Scratch is one of them where kids learn to program using visual blocks. However, most users opt for written languages because these enable students to learn programming languages like Python or C++ much more quickly.

There are a few attempts at Bangla Programming Language. Dr. Nova Ahmed, Associate Professor at North South University, had done one of the notable projects named "ছ Script."[02] She used the ECMA Script as Grammar and Jison to parse the grammar, and at the end, she used JavaScript as the interpreter. An interpreter like JavaScript is widely executable, but code execution becomes slower. Using a compiler instead of an interpreter will make it faster. Another Programmer from India, Palash Bauri, had done another project named "The Pankti Programming Language" based

on a compiler, but he used Zig language.[01] This language is complex and advanced, which is unsuitable for secondary-level students.

So, here, we used C++ language to build the compiler, which can run a program faster than a typical interpreter. Furthermore, we kept it as simple as Dr. Nova Ahmmed did with some extra flexibility of data type casting and dynamic array which can hold any types of data.

Methods:

The steps to build the compiler are given below:

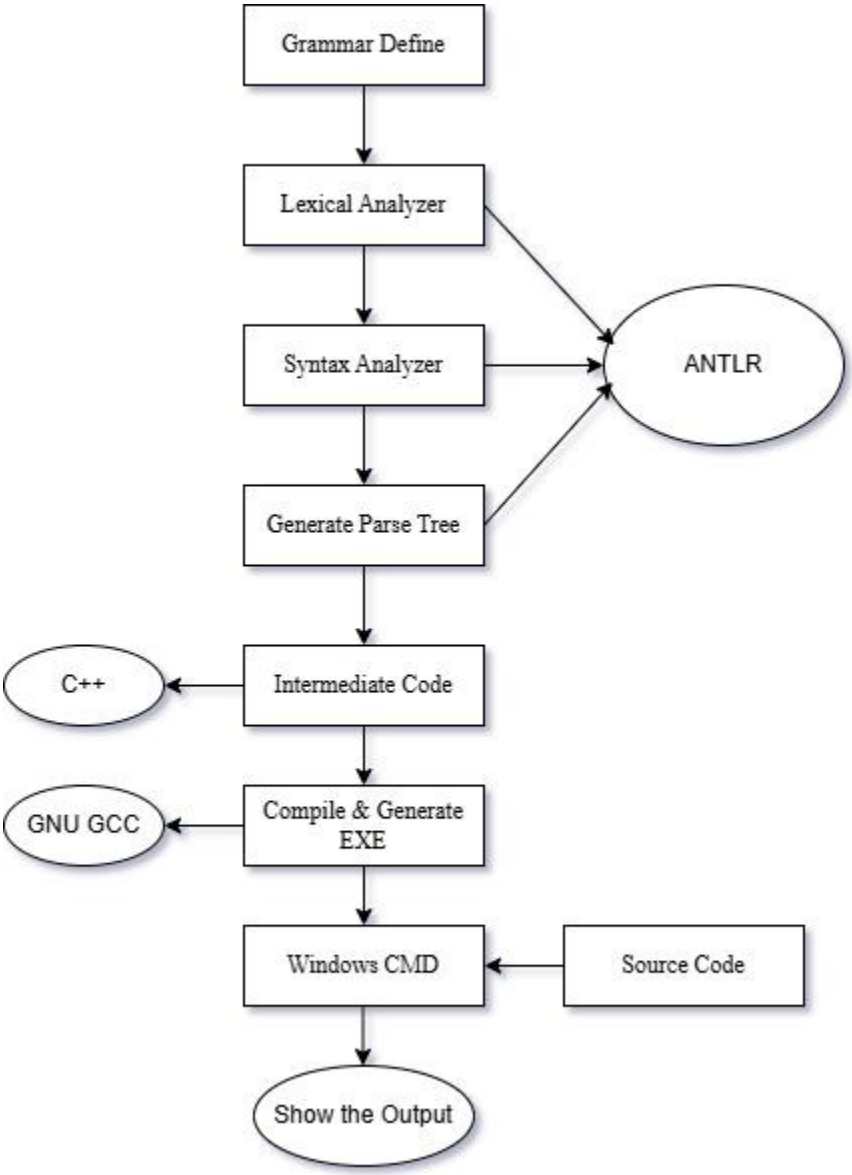

Figure 1: Flowchart of our Project

As shown in the diagram above, we first construct the grammar according to our needs. We added the features one by one to our compiler. The complete grammar of our programming language can

be viewed from the GitHub repository.[05] After constructing the grammar, we set up the ANTLR and generated the Lexical Analyzer and the parse tree. Then with the help of AI such as Copilot we learned about the generated lexical analyzer code about how to call these objects and use for implementing the visiting pattern. In this step, we implemented the listener or visiting pattern and used Codeblocks to compile the project. After compiling the project we get an executable exe file, we need to run the exe file on the windows CMD with the source code as an argument. Then, we can see the output on the terminal. A complete source code that can demonstrate all the features of our compiler is available on the repository.[06]

API: We build an API based on python using flask and html. In this API we used a python package named "subprocess" to run the command on a windows shell and capture the output in Unicode-8 with a timeout of 15 seconds to stop the infinite loop. Timeout used for the debugging purposes during the implementation of "for loop", so that we can stop the process if its start running an infinite loop.

Challenges, and remedies used:

During the implementation of this project, we faced a lot of problems, and we chose many alternatives to complete the project. The first challenge was to show the Bangla Unicode characters on the terminal. Many alternatives were available, one of them was to install Bangla fonts such as Bijoy or Avro. We tried Bijoy but it was not working, we don't know the reason. Then we find the easiest way to show the Bangla Characters as output is to build an API and use html as GUI of the API. Now we can type our source code on a web framework using a browser and it can show the output using the python API. Second challenge was to support nested if-else statement. As our previous report, we were unable to implement the nested if-else statement. After using some flags to count the inner blocks, we successfully implemented the nested if-else statement. As our nested if-else was working perfectly we just followed the same step to construct the nested for loop statement. As we are adding features one by one and out compiler getting complex to handle, as a result we faced some issue regarding the array implementation. Data structures are causing some problems. If we change the data structure that holds the variables name and value, to support the array, we need to change the entire project structure. To solve this problem, we just declare another global variable to hold the arrays name and values. And finally, we successfully added the features of 1D array.

Unsolved challenges and potential solutions:

We tried to add some simple file handling features, we constructed the grammar, generated the lexical analyzer, also implemented the visiting pattern; after running the project, our compiler was unable to create new file or read data from a file. We tried many debugging steps to find out the problem, and we found that the windows file system ran into a bad state just after opening a file, which caused the problem to read or write data in a file. It might be a potential problem, and we were unable to find the solution as time is running out for submission. We have got enough time for this project but unfortunately for many reasons and academic purposes we were unable to build all the features required to build a compiler for beginners. One important feature is to take input from user. We were unable to build it as we couldn't solve the problem to show the Bangla

Unicode characters on the terminal. Taking input from the API is also possible, we need to learn that too.

Conclusion:

To sum up, our project shows how a Bangla programming language could help secondary school students in Bangladesh overcome their challenges when learning to program. We have taken crucial steps in making programming more approachable and relatable for young learners by creating a compiler-based solution that combines basic programming concepts with native Bangla syntax. Although conditionals, loops, and arrays are essential components of our current system, they also offer a strong basis for upcoming additions like file management, user input handling, and support for more complex programming structures.

Our efforts show the feasibility and significance of localized programming languages in promoting computational thinking despite obstacles like Unicode character display and project scope limitations brought on by time constraints. These skills can be improved in further versions of this project, increasing the effect and inclusivity of programming instruction. Through ongoing development and cooperation, this project encourages wider acceptance and innovation in local-language programming tools. Complete Project Code can be found on GitHub Repository. [07]

Contribution Table:

Group Member	Contribution
Joy Kumar Ghosh – 2211424 6 42	01. Literature Review of Official Proposal & Final Report 02. Methodology of Report 1 and Final Report 03. Complete Coding Part
Ahanaf Tahomid – 1831920 6 42	01. First Unofficial Proposal 02. Second Official Proposal 03. Support on Report 1 and Final Report

Signature:

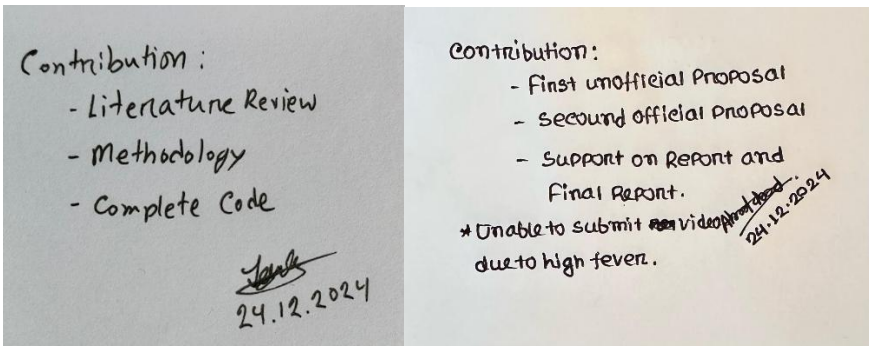

Figure 2:Proved of Contribution

References:

01. Palash Bauri, The Pankti Programming Language, Last modified August 18, 2024. https://github.com/bauripalash/pankti
02. Ahmed, Nova, Arman Kamal, Adnan Nuruddin, and Syed Tanveer Jishan. "My code in my native tone: Cha Script." In *Proceedings of the Eighth International Conference on Information and Communication Technologies and Development*, pp. 1-4. 2016.
03. OpenAI, ChatGPT, 2024. https://chatgpt.com/
04. Microsoft, GitHub Copilot – GPT 4.0, 2024. https://github.com/features/copilot
05. Joy Kumar Ghosh, Bangla Programing Language, Last Modified December 24, 2024. https://github.com/LTJ508/Bangla_Programing_Language/blob/main/BanglaProgramingLanguage/ANTLR_Generated/Bangla.g4
06. Joy Kumar Ghosh, Bangla Programing Language, Last Modified December 24, 2024. https://github.com/LTJ508/Bangla_Programing_Language/blob/main/BanglaProgramingLanguage/Source.bangla
07. Joy Kumar Ghosh, Bangla Programing Language, Last Modified December 24, 2024. https://github.com/LTJ508/Bangla_Programing_Language/tree/main