North South University

Department of Electrical & Computer Engineering

Project Proposal

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: October 19, 2024

Submitted by:

   01. Ahanaf Tahomid – 1831920 6 42
   02. Joy Kumar Ghosh – 2211424 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 are trying to assemble some fundamental operations, including assignments, conditionals, loops, arrays, and simple file handles using the complete Bangla Language.

Furthermore, we are trying to keep it more orthogonal so that users can define identifiers and numbers using both Bangla and English literals but can't combine them into one identifier or number. Many programmers in India and Bangladesh have built the Bangla Programming Language, but most languages are built based on interpreters like Java. Palash Bauri is one of them who built a Bangla Programming Language based on a compiler, but he used Zig language to implement it.[01] Here, we are going to use C++ language to complete this project. Also, this build will be more straightforward for a secondary student to understand.

Introduction and Motivation:

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.

Background and significance:

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.

But 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 "চা 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 will use C++ language to build the compiler, which can make a faster executable file. Furthermore, we will keep it as simple as Dr. Nova Ahmmed did.

Method and Design:

Main objectives of this project will be like this:

C	Probable Source Code of this project

#include <stdio.h>
int main() {
  printf("Hello, World!\n");
  return 0;
}
	শুরু
	  দেখাও ("হ্যালো ওয়ার্ল্ড")
	শেষ

Output: Hello, World	Output: হ্যালো ওয়ার্ল্ড

Block Diagram of the Compiler Design: The Bengali programming language compiler will follow a compiler's standard structure, with specific adaptations for handling Bengali keywords and syntax. The following block diagram illustrates the compiler design:
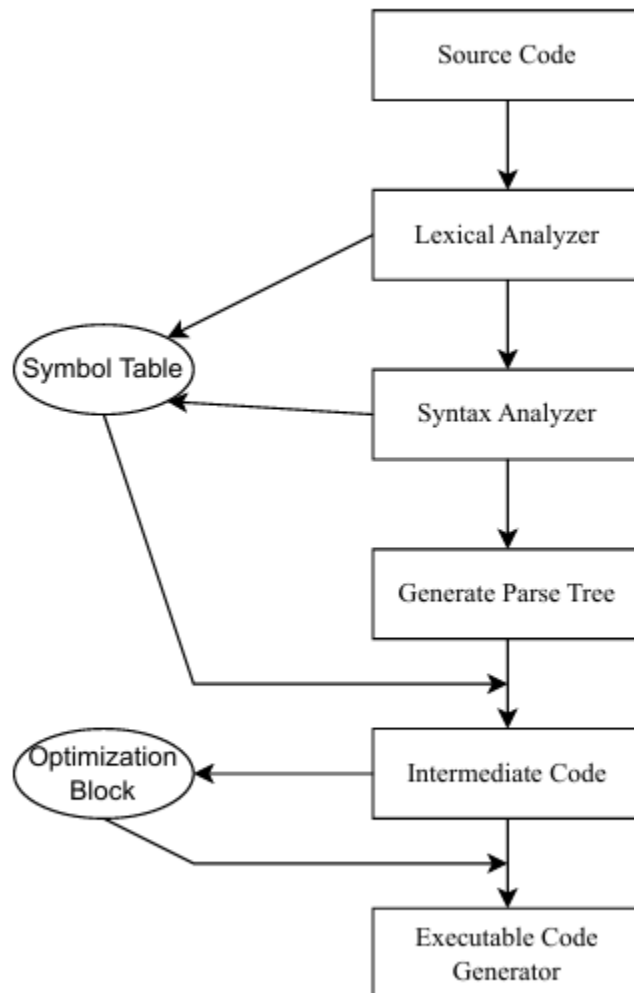

Figure 1: Compiler Diagram, Created with Draw.io

Lexical Analyzer: Receives tokens from the lexical analyzer and constructs a parse tree to ensure that the code follows the grammar rules of the programming language. It detects syntax errors and reports them for correction.

Probable tokens:

('IF', r'যদি'),   # If: যদি

('ELSE', r'নাহলে'),   # Else: নাহলে

('NUMBER', r'[0-9০-৯]+'),   # Numbers: ০ to ৯ (Bangla digits)

 ('ID',   r'[a-zA-Zঅ-ঔক-হ_][a-zA-Zঅ-ঔক-হ_0-9]*'), #Identifiers(Bangla variable names)

('EQ', r'='),   # Equals: =

('PLUS', r'\+'),   # Addition: +

('MINUS', r'-'),   # Subtraction: -

('LPAREN', r'\('),   # Left parenthesis: (

('RPAREN', r'\)'),   # Right parenthesis:)

('LBRACE', r'\{'),   # Left brace: {

('RBRACE', r'\}'),   # Right brace: }

('SEMICOLON', r';'),   # Semicolon: ;

('LT', r'<'),   # Less than: <

('GT', r'>'),   # Greater than: >

- **Syntax Analyzer:** Checks for semantic correctness, ensuring that operations are meaningful, like type compatibility and correct variable usage according to the Grammar Construct. Generate Parse Tree (AST).

- **Intermediate Code Generator:** Transforms the high-level code into an intermediate representation (IR) that is abstract yet closer to machine code. This code is easy to optimize and acts as a bridge between the source code and machine code.

- **Executable Code Generator:** With the help of Linker, this module built an executable binary code.

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.