

North South University
Department of Electrical & Computer Engineering
Project Report - 1

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: November 29, 2024

Submitted by:

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

Submitted to:

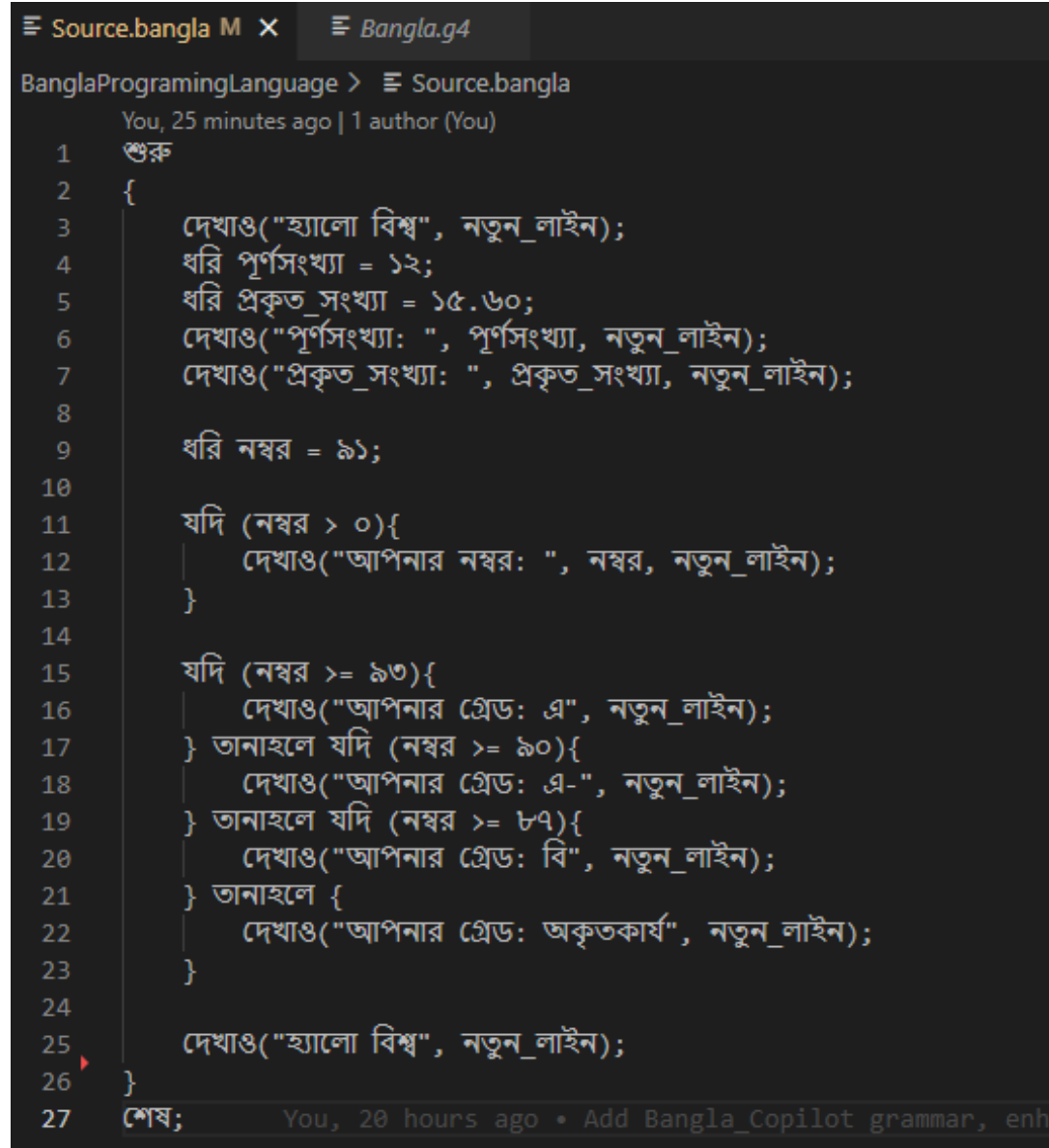
Course Instructor, Dr. Md Shahriar Karim (MSK1)

Project Aims:

Our project aims to develop a Bangla Programming Language for secondary students. As Bangla is our mother language and most of the student's study in Bangla, we decided to construct a compiler that can execute Bangla Program.

Methodology:

Source Code: For construct a grammar, first we need to demonstrate a simple source code, which represents our needs to design the grammar.



```
Source.bangla M X Bangla.g4
BanglaProgrammingLanguage > Source.bangla
You, 25 minutes ago | 1 author (You)
1 শুরু
2 {
3     দেখাও("হ্যালো বিশ্ব", নতুন_লাইন);
4     ধরি পূর্ণসংখ্যা = ১২;
5     ধরি প্রকৃত_সংখ্যা = ১৫.৬০;
6     দেখাও("পূর্ণসংখ্যা: ", পূর্ণসংখ্যা, নতুন_লাইন);
7     দেখাও("প্রকৃত_সংখ্যা: ", প্রকৃত_সংখ্যা, নতুন_লাইন);
8
9     ধরি নম্বর = ৯১;
10
11     যদি (নম্বর > ০){
12         দেখাও("আপনার নম্বর: ", নম্বর, নতুন_লাইন);
13     }
14
15     যদি (নম্বর >= ৯০){
16         দেখাও("আপনার গ্রেড: এ", নতুন_লাইন);
17     } তানাহলে যদি (নম্বর >= ৯০){
18         দেখাও("আপনার গ্রেড: এ-", নতুন_লাইন);
19     } তানাহলে যদি (নম্বর >= ৮৭){
20         দেখাও("আপনার গ্রেড: বি", নতুন_লাইন);
21     } তানাহলে {
22         দেখাও("আপনার গ্রেড: অকৃতকার্য", নতুন_লাইন);
23     }
24
25     দেখাও("হ্যালো বিশ্ব", নতুন_লাইন);
26 }
27 শেষ; You, 20 hours ago • Add Bangla Copilot grammar, enh
```

Figure 1: Bangla Source Code - Hello World

Then we started to design the grammar to support this source code.

Grammar Design: First step to construct a compiler is to design a grammar. As we already demonstrated a source code for our programming language, by keeping focus on that we design our grammar. Some screenshots are given below:

```
// Lexer Rules
SHURU: 'শুরু';
SHESH: 'শেষ';
DHORI: 'ধরি';
DEKHAO: 'দেখাও';
NATUN_LINE: 'নতুন_লাইন';
IF: 'যদি';
ELSE_IF: 'তনাহলে যদি';
ELSE: 'তনাহলে';

ID: ID_START ID_PART*; // Identifier

fragment ID_START: [অ-ঈ]; // First character of identifier (excluding '০ং')
fragment ID_PART: [অ-ঈ০-৯০ং]; // Subsequent characters of identifier

FLOAT: [০-৯]+.'[০-৯]+; // Floating-point number
INT: [০-৯]+; // Integer number
STRING: '"' .*? '"'; // String literal

WS: [ \t\r\n]+ -> skip; // Whitespace
```

Figure 2: Lexer Rules of our Grammar

```
Source.bangla M Bangla.g4 X
BanglaProgrammingLanguage > ANTLR_Generated > Bangla.g4 > ...
You, 27 minutes ago | 1 author (You)
1 grammar Bangla;
2
3 // Parser Rules
4 program
5 : SHURU block SHESH
6 ;
7
8 block
9 : '{' statement* '}'
10 ;
11
12 statement
13 : variableDeclaration
14 | printStatement
15 | ifStatement
16 ;
17
18 variableDeclaration
19 : DHORI ID '=' (INT | FLOAT) ';'
20 ;
21
22 printStatement
23 : DEKHAO '(' printArguments ')' ';'
24 ;
25
26 printArguments
27 : (ID | STRING) (',' (ID | STRING))* (',' NATUN_LINE)?
28 ;
29
```

```
29
30 ifStatement
31 : IF '(' condition ')' block (ELSE_IF '(' condition ')' block)* (ELSE block)?
32 ;
33
34 condition
35 : operand comparisonOperator operand
36 ;
37
38 operand
39 : ID
40 | INT
41 | FLOAT
42 ;
43
44 comparisonOperator
45 : '>='
46 | '<='
47 | '>'
48 | '<'
49 | '=='
50 | '!='
51 ;
```

Figure 3: Parser Rules of our Grammar

Parser & Lexer: The next step for constructing a compiler is to generate or define the Parser and Lexer Rules. In this step, we used ANTLR to generate the lexer and parser rules. The environment setup for ANTLR was so difficult for us. After a long search on google and YouTube we didn't find any relevant tutorial for ANTLR Cpp environments. Then we decided to take help from the ChatGPT, and after a long conversation, we solve the environment problem and successfully generated the Lexer and Parser.

IR Code Generator: ANTLR generated a bunch of files, among them one was BanglaBaseListener.h, where abstract function of required entering and existing pattern that means AST Tree visiting patterns were defined. So, now we need to implement these functions according to our needs. In this step we take helps of Copilot to implement these functions according to our needs. And after a long error and fix trial we successfully implemented these functions and it's working now.

API: When we implemented the visiting pattern functions, we tried to run our compiler through windows command prompt. But we encountered some problems that the terminal was unable to show Bangla Unicode Characters. Then we decided to build a Python API to run this compiler on a web framework. And we successfully designed an API to run this compiler and introduced to a new problem, that our compiler was unable to assign a variable as Bangla Unicode is not supported by Cpp. Then we implemented some helper functions which convert the Bangla Unicode to English Unicode during the variable assignment and convert back to the Bangla Unicode during the printing instruction. Screenshot of our API is given below:

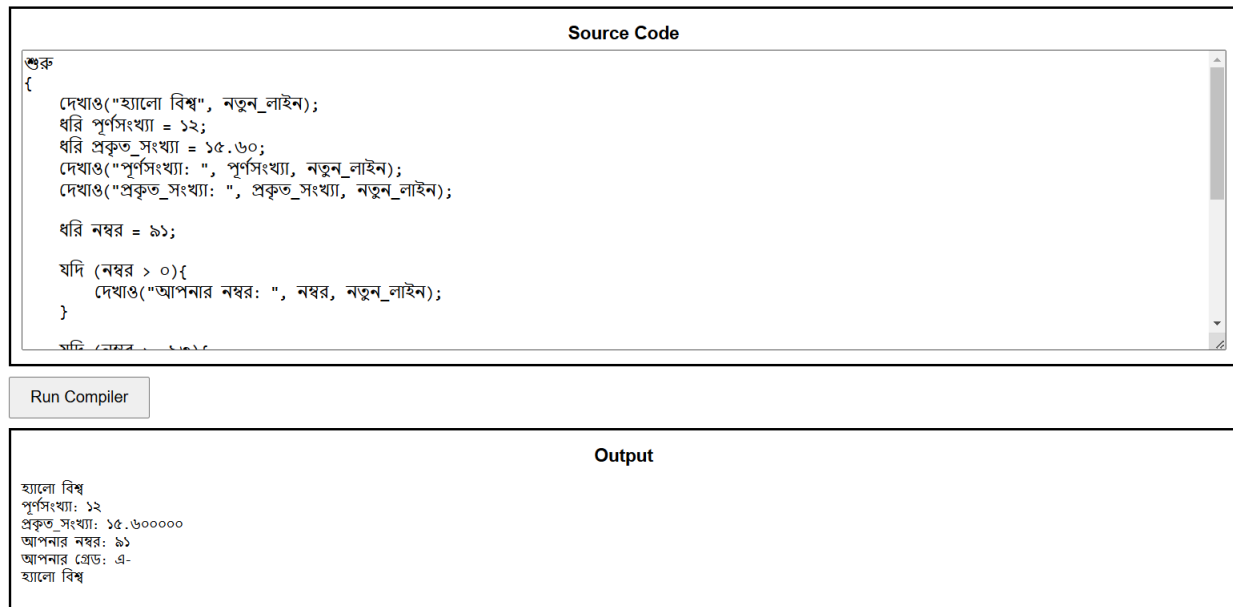


Figure 4: Bangla Programming Language - API

Bugs & Future Work: So far, we have added support of variable assignment, printing string and identifier, and conditional statement. But, in the conditional statement, we are facing some issue regarding the nested "IF-ELSE" statement. We added some debugging statements to find out the error. And we got this:

Source Code

```
ধরি ক = ৫;
ধরি খ = ১০;
ধরি গ = ৪;
যদি (ক >= খ){
  যদি (ক >= গ){
    দেখাও("সবথেকে বর সংখ্যা, ক১: ", ক, নতুন_লাইন);
  } তনাহলে{
    দেখাও("সবথেকে বর সংখ্যা, গ২: ", গ, নতুন_লাইন);
  }
} তনাহলে যদি (খ >= গ){
  দেখাও("সবথেকে বর সংখ্যা, খ৩: ", খ, নতুন_লাইন);
} তনাহলে{
  দেখাও("সবথেকে বর সংখ্যা, গ৪: ", গ, নতুন_লাইন);
}
```

Run Compiler

Output

```
Debug => Evaluating condition: 5 >= 4 -> true
Debug => Processing statement: দেখাও("সবথেকে বর সংখ্যা, ক১: ", ক, নতুন_লাইন);
সবথেকে বর সংখ্যা, ক১: ৫
Debug => Evaluating condition: 5 >= 10 -> false
Debug => Evaluating condition: 10 >= 4 -> true
Debug => Processing statement: দেখাও("সবথেকে বর সংখ্যা, খ৩: ", খ, নতুন_লাইন);
সবথেকে বর সংখ্যা, খ৩: ১০
```

That means our visiting pattern is currently evaluating the inner statement first then evaluating the outer statement. We will try to fix this in future or before the final report, also we will add some more features like “FOR” loop, and some simple “FILE” operation.

Conclusion:

Programming in Bangla Language will really help to understand the mechanism by the secondary level students. Although we have constructed a very little part of a compiler, in future we can extend this compiler to support more features.

GitHub Repository:

https://github.com/LTJ508/Bangla_Programing_Language

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>