**NIHAO –Milestone 2**

Our team has decided to name the compiler ‘Nihao’ which is Hello in Chinese. The inspiration behind the name comes from one of our team member Jiuxu.

Designing the compiler consists of significant stages and we have fairly distributed the work of every stage between our different team members.

The first stage involves defining the structure of the language and providing meaning to it, also known as grammar

For defining our grammar we’ve used Extended Backus-Naur Form of Context Free Grammar. The Lexical Structure of our Language consists of following:

Reserved Keywords: while, do, od, if, start, end

Constants = 0,1,2,3,4,5,6,7,8,9

Special Symbols = ‘+’ , ‘-‘ , ‘/’, ‘\*’, ‘>’, ‘<’, ‘<=’, ‘>=’, ‘==’, ‘%’

Identifiers = ‘a’ , ‘b’ …….. ‘z’

The syntax of our grammar supports the Data Type **Int,** Decision Control Statement **If–then and Else,** Assignment Operator **‘=’, While** Loop for iterative execution. The syntax does not support curly braces to mark the beginning and end of the loops, instead we have used the keywords ‘**do**’ and ‘**od**’ for while loop and ‘**start**’ & ‘**end**’ for if –then else construct which mark the beginning and end of it.

Our language supports addition, subtraction, multiplication and division as the arithmetic operations and supports the order of precedence.

We’ve defined the syntax of our language to be a program which consists of a statement set which might terminate on one statement or further have a statement set. We’ve implemented recursion in our grammar for this purpose.  
The statement further supports an assignment statement, an iterative statement or a decision control statement.

The iterative statement , While loop consists of an expression, the keyword “do” and “od” which mark the beginning and end of the loop and similarly for If loop defined with the keyword “start” and “end”.