****

**DHA SUFFA UNIVERSITY**

**Department of Computer Science**

**CS-313**

**Compiler Construction**

**Fall 2019**

**Urdu Script**

**Language Specification**

**Submitted to: Ms. Raazia Sosan**

**GROUP MEMBERS**

Bahawal Baloch cs171032

Mohammad Raza cs171064

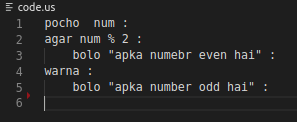
**DATE: 7th JANUARY 2020**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Index** | **Page** | **Page Number** |
| 1. | Title Page | 1 |
| 2. | Table of Contents | 2 |
| 3. | Motivation | 3 |
| 4. | Code example | 3 |
| 5. | Tokens | 3 |
| 6. | Flex Code | 4 |
| 7. | Flex out | 4 |
| 8. | C++ lex output | 5 |
| 9. | C++ parser output | 5 |

**Motivation:-**

Computers have become an imminent need of our society. In the future, computer programing would be essential for everyone for it helps in developing problem solving skills. One of the biggest flaws of our school systems is that they are unable to teach programming to children due to lack of teaching abilities. Another factor I see causing hindrance in the learning is the english syntax and grammar. So I am going to make a miniature programming language that will be in Roman Urdu which will overcome the language barrier.This language will have minimal keywords that can be used to solve simple problems.This language we intend to spark the passion of computers in young minds and give them the magic of programming.



**KEYWORDS:(all keywords will be in lowercase)**

**bolo** Command to print variable.

**pocho** Command to take input.

**agar**

Equivalent of if.

**Warna**

Equivalent of else

**dubara N**

Command to repeat a below code n times

**ruko**

Command to break loop

**Khatam**

Command to terminate task/program

**Kaam**

Keyword to declare a function

%{

#include <stdio.h>

#include <stdlib.h>

%}

keyword bolo|pocho|dobara|agar|warna|ruko|khatam|kam

letter [a-zA-Z]

number -?[0-9](\.[0-9]+)?

string \"([^\\\"|\\.])\*\"

assignment \=

sum \+

sub \-

operator {sum}|{sub}|{assignment}

identifier [letter]+

%%

{string} {printf("%s is a string\n",yytext);}

{operator} {printf("%s is a operator\n",yytext);}

{number} {printf("%s is a Number\n",yytext);}

{keyword} {printf("%s is a Keyword\n",yytext);}

{identifier} {printf("%s is a identifier\n",yytext);}

%%

int yywrap(){}

int main(){

FILE \*fp;

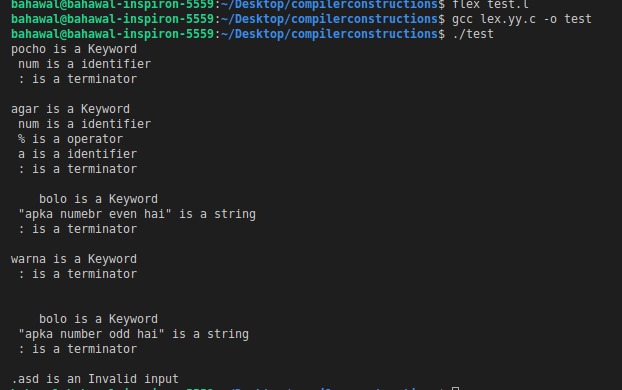
fp = fopen("humarilanguage.txt","r");

yyin = fp;

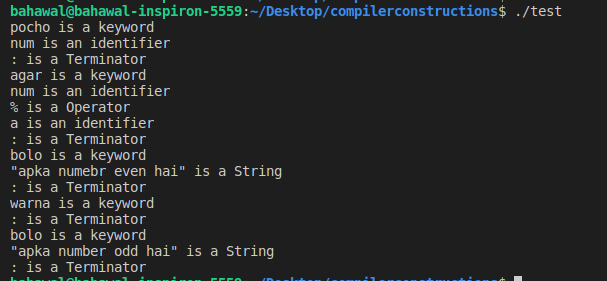
yylex();

return 0;

}



DFA OUTPUT



PARSER OUTPUT

