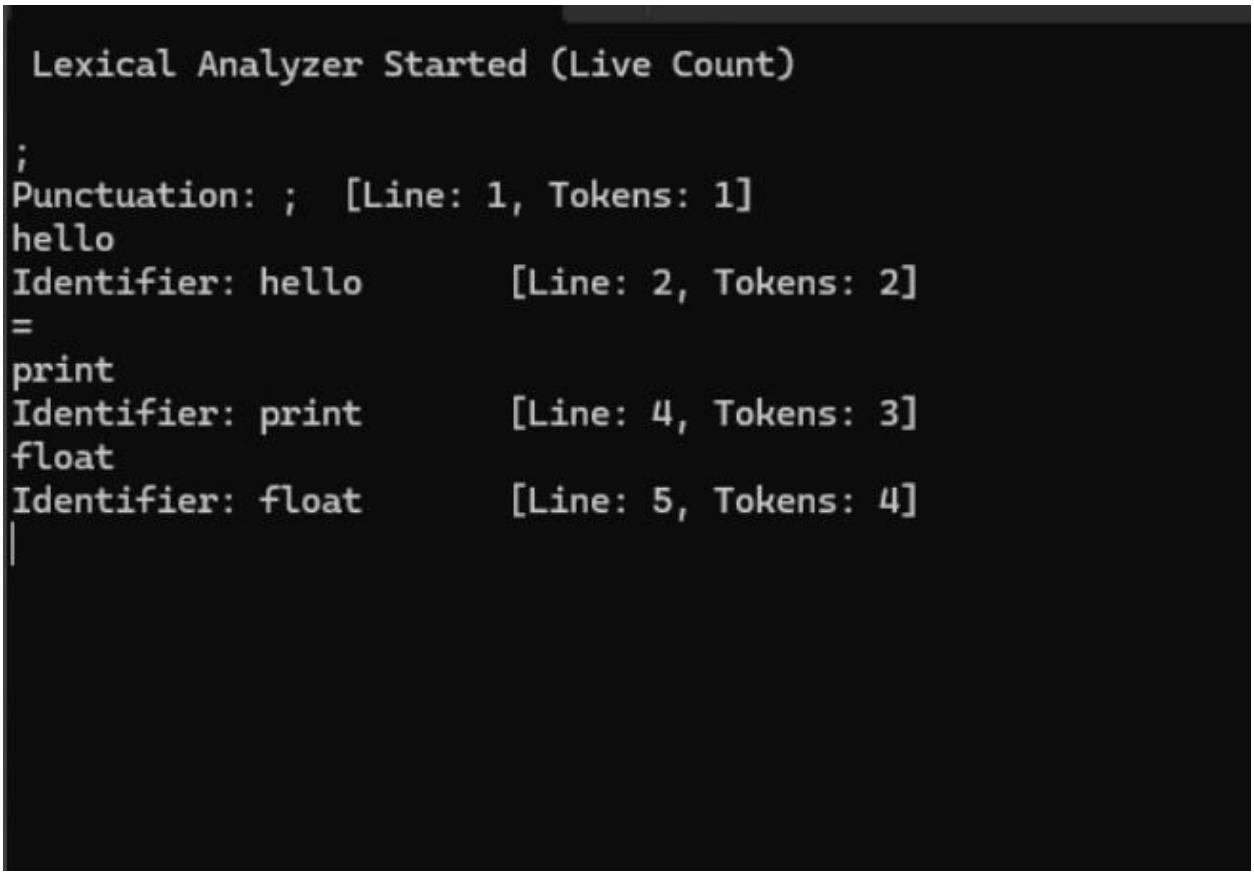
Output :


Lexical Analyzer Started (Live Count)
;
Punctuation: ; [Line: 1, Tokens: 1]
hello
Identifier: hello [Line: 2, Tokens: 2]
=
print
Identifier: print [Line: 4, Tokens: 3]
float
Identifier: float [Line: 5, Tokens: 4]

Question 2: Keywords, Meanings, and Example Usage

Keyword	Meaning	Example Usage
initial	Declares an integer variable	`int age = 25;`
pointed	Declares a floating-point variable	`float price = 9.99;`
point var	Declares a double-precision floating-point variable	`double pi = 3.14159;`
alphabet	Declares a character variable	`char grade = 'A';`
truse	Represents a boolean (true/false) value	`bool isReady = true;`
then	Used for conditional branching	`if (x > 0) cout << "Positive";`
swap	Multi-way branch based on value	`switch (day) { case 1: cout << "Mon"; break; }`
suit-case	Defines a branch in switch	`case 2: cout << "Tue"; break;`
breakUp	Exits a loop or switch block	`if (x == 5) break;`
continued	Skips current loop iteration	`if (x == 5) continue;`
form	Loop that repeats a block of code	`for (int i=0; i<5; i++)`
white	Exists from a functional and operational value	`return 0;`
structure	Groups variables under one name	`struct Student { int id; string name; };`
uni-class	Defines an object-oriented class	`class Car { public: void start(); };`
@all	Access specifier; members accessible outside class	`public: int speed;`

Operators

Operator	Meaning	Example Usage
+	Addition operator	`int sum = a + b;`
-	Subtraction operator	`int diff = a - b;`
/	Division operator	`int div = a / b;`
*****	Multiplication operator	`int product = a * b;`
%	Modulus operator	`int rem = a % b;`

Punctuation

Punctuation	Meaning	Example Usage
;	Statement terminator	`cout << "Hello";`
,	Separator for values or parameters	`int x = 5, y = 10;`
{}	Used to define code blocks	`cout << "Inside block";`
[]	Array subscript or indexing	`arr[2] = 10;`

Regular Expressions

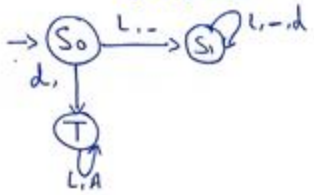
Token Type	R.E (Regular Expression)	Example
Pointed	[0-9]*\.[0-9]*	`1.23`
Identifier	[a-zA-Z_][a-zA-Z0-9_]*	`count1`
String	"[a-zA-Z0-9]*"	`"Zunair"`
Keyword	(funct)	`Num`
Operator	(\+)	`(no example given)`

Name: Registration No.

Question 5

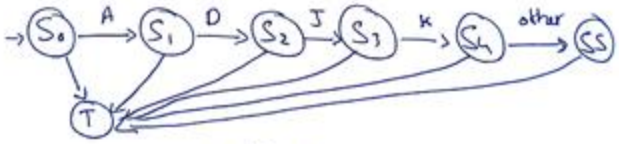
Finite Automata State Diagrams

(i.) Identifier

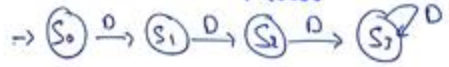


A = Alphabet
D = digit
- = spaces.
T = Trap.

(ii.) Keyword



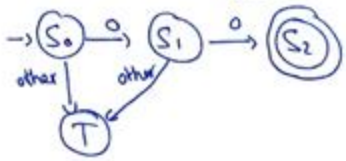
(iii.) Float



Name:

(iv.)

Operator



O = Operator

(v.)

Comments