RA1911033010126
BVS Satwika

LAB EXP1: LEXICAL ANALYSIS

AIM:

To write a program to implement Lexical Analysis.

ALGORITHM:

- Start.
- Get the input program from the file lex.txt.
- Create lists of Special character, Keyword, Operator, Header files, Identifier, Integer.
- If the word is in any of the above lists, append it to a separatelist and repeat this step till the last line of the lex.txt file.
- Print the identifiers/tokens and their respective counts inthe lex.txt file.
- Stop.

CODE:

```
//RA19110330126
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<ctype.h>
#include<vector>

int isKeyword(char buffer[]){
char keywords[37][10] =
{"auto","using","namespace","include","break","case","char","const","continu
e","default",
"do","double","else","enum","extern","float","for","goto",
"if","int","long","public","register","return","short","signed",
"sizeof","static","struct","switch","typedef","union",
"unsigned","void","volatile","while","string"};

int i, flag = 0;
for(i = 0; i < 37; ++i){
if(strcmp(keywords[i], buffer) == 0){
flag = 1;
```

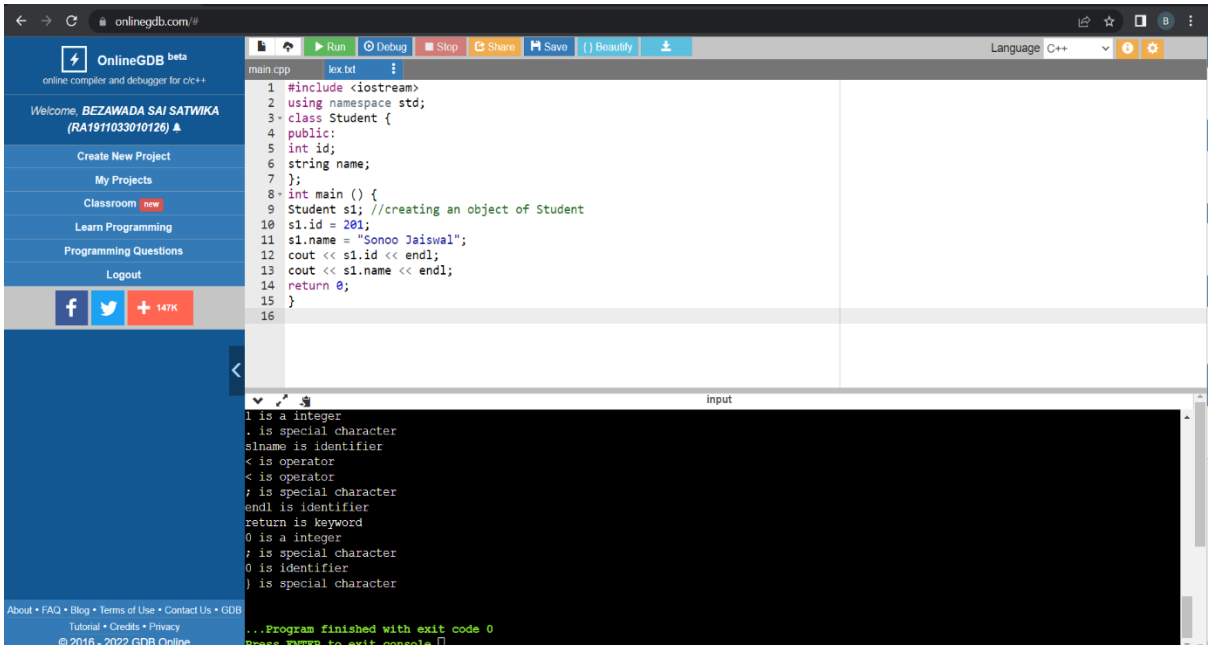
```
break;
}
}
return flag;
}

char isHeader(char buffer[]){
   char headers[2][10] = {
      "iostream", "stdio"
   };

int i, flag = 0;
for(i = 0; i < 2; ++i){
if(strcmp(headers[i], buffer) == 0){
flag = 1;
break;
}
}
return flag;
}

int main(){
char ch, buffer[15], operators[] = "+-*/%=<>", special[] = "#;,.{}[]()";
FILE *fp;
int i,j=0;
fp = fopen("lex.txt","r");
if(fp == NULL){
printf("error while opening the file\n");
exit(0);
}
while((ch = fgetc(fp)) != EOF){
  for(i = 0; i < 8; ++i){
  if(ch == operators[i])
  printf("%c is operator\n", ch);
  }

  for(i = 0; i < 10; ++i){
  if(ch == special[i])
  printf("%c is special character\n", ch);
  }
```

```
  if(isdigit(ch)==true)
  printf("%c is a integer\n",ch);

  if(isalnum(ch)){
  buffer[j++] = ch;
  }
  else if((ch == ' ' || ch == '\n') && (j != 0)){
  buffer[j] = '\0';
  j = 0;

  if(isKeyword(buffer) == 1)
  printf("%s is keyword\n", buffer);
  else if(isHeader(buffer)==1)
  printf("%s is Header file\n", buffer);
  else
  printf("%s is identifier\n", buffer);
  }

}
fclose(fp);
return 0;
}
```

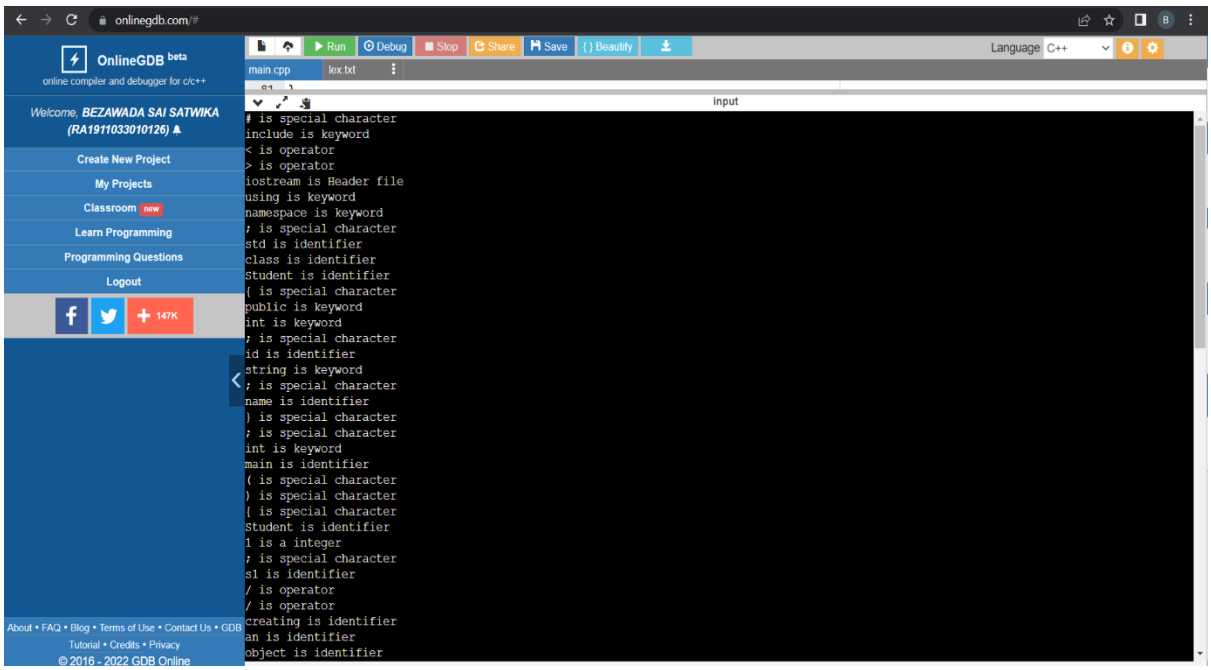
lex.txt:

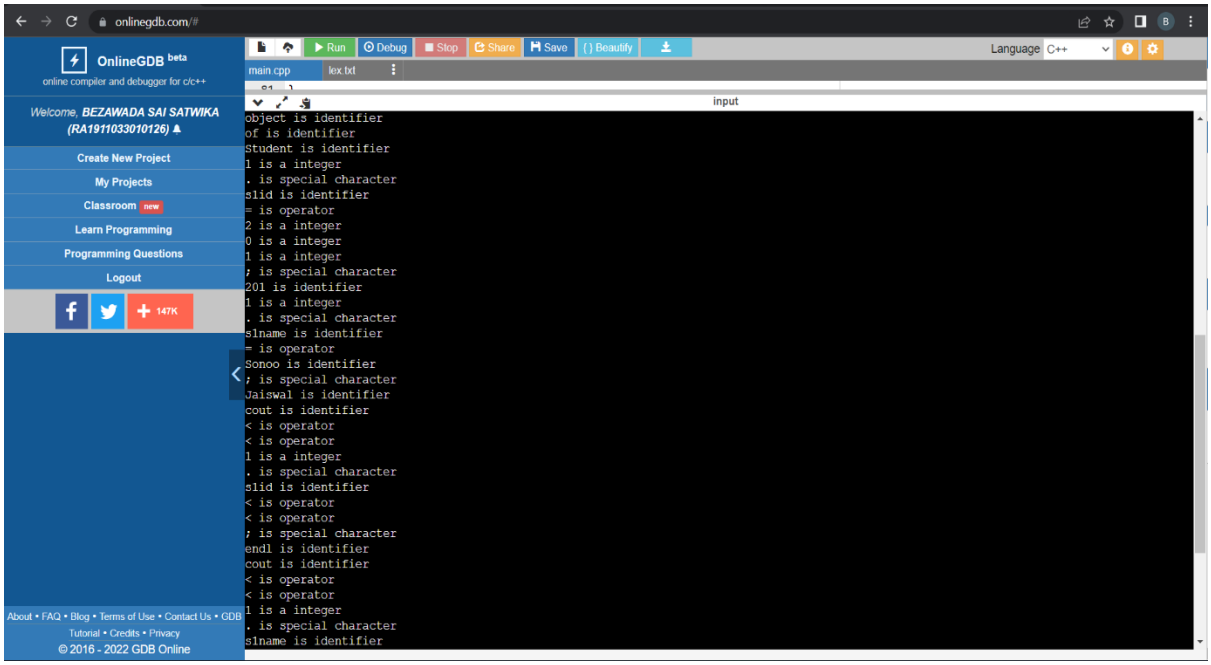
```
#include <iostream>
using namespace std;
class Student {
public:
int id;
string name;
};
int main () {
Student s1; //creating an object of Student
s1.id = 201;
s1.name = "Sonoo Jaiswal";
cout << s1.id << endl;
cout << s1.name << endl;
return 0;
```

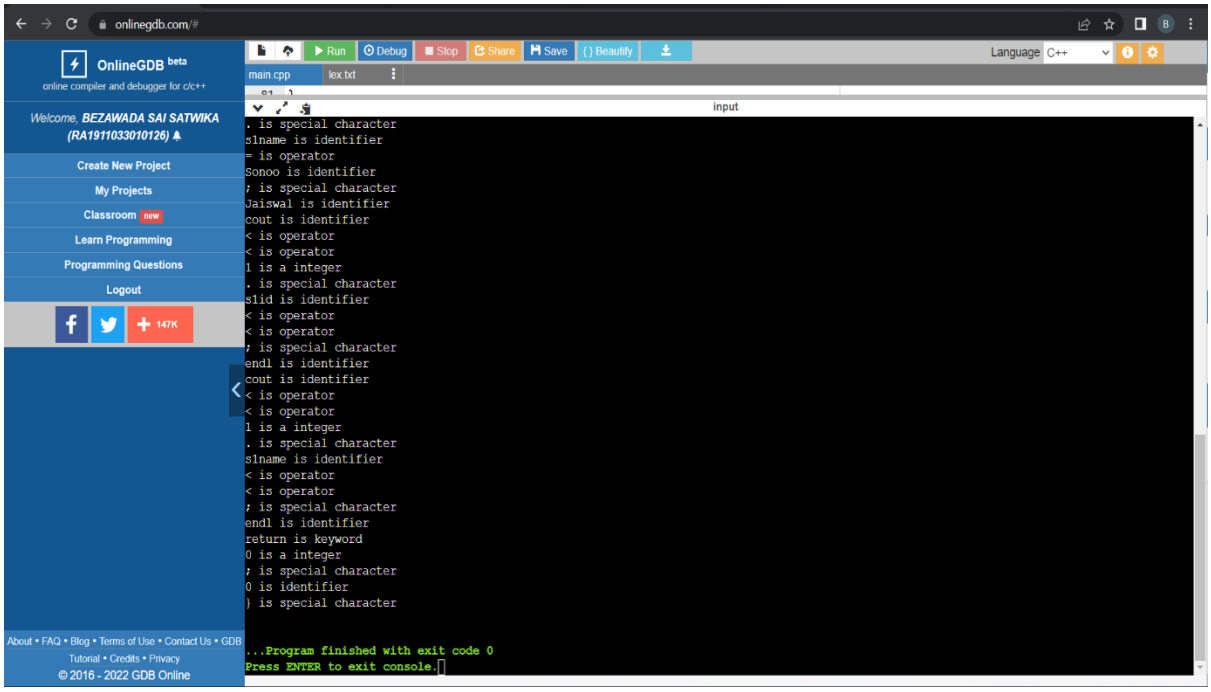
}


onlinegdb.com/#
OnlineGDB beta
online compiler and debugger for c/c++
Welcome, BEZAWADA SAI SATWIKA (RA1911033010126)
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Logout
147K
Run
Debug
Stop
Share
Save
{ } Beautify
Language C++
main.cpp
lex.txt
1 #include <iostream>
2 using namespace std;
3 class Student {
4 public:
5 int id;
6 string name;
7 };
8 int main () {
9 Student s1; //creating an object of Student
10 s1.id = 201;
11 s1.name = "Sonoo Jaiswal";
12 cout << s1.id << endl;
13 cout << s1.name << endl;
14 return 0;
15 }
16
input
1 is a integer
. is special character
s1name is identifier
< is operator
< is operator
; is special character
endl is identifier
return is keyword
0 is a integer
; is special character
0 is identifier
} is special character
...Program finished with exit code 0
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2022 GDB Online

OUTPUT:


onlinegdb.com/#
OnlineGDB beta
online compiler and debugger for c/c++
Welcome, BEZAWADA SAI SATWIKA (RA1911033010126)
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Logout
147K
Run
Debug
Stop
Share
Save
{ } Beautify
Language C++
main.cpp
lex.txt
input
is special character
include is keyword
< is operator
> is operator
iostream is Header file
using is keyword
namespace is keyword
; is special character
std is identifier
class is identifier
Student is identifier
{ is special character
public is keyword
int is keyword
; is special character
id is identifier
string is keyword
; is special character
name is identifier
} is special character
; is special character
int is keyword
main is identifier
(is special character
) is special character
{ is special character
Student is identifier
1 is a integer
; is special character
s1 is identifier
/ is operator
/ is operator
creating is identifier
an is identifier
object is identifier
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2022 GDB Online





RESULT:

The implementation of lexical analysis is done using the code.