

PROGRAM CODE:

In C++

```
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
```

```
using namespace std;
```

```
int iskeyword(char buffer[]) {
    char keywords[32][10] = { "auto", "break", "case", "char",
    "const", "continue", "default", "do", "double", "else", "enum",
    "extern", "float", "for", "goto", "if", "int", "long",
    "register", "return", "short", "signed", "sizeof",
    "static", "struct", "switch", "typedef", "union", "void",
    "unsigned", "volatile", "while" };
}
```

```
int i, flag=0;
for (i=0; i<32; ++i) {
    if (strcmp(keywords[i], buffer) == 0) {
        flag = 1;
        break;
    }
}
return flag;
}
```

```
int main() {
    char ch, buffer[15], operators[] = "+-*/%=";
    ifstream fin("program.txt");
    int i, j=0;
    if (!fin.is_open()) {
        cout << "error while opening the file \n";
    }
}
```

Input:program.txt

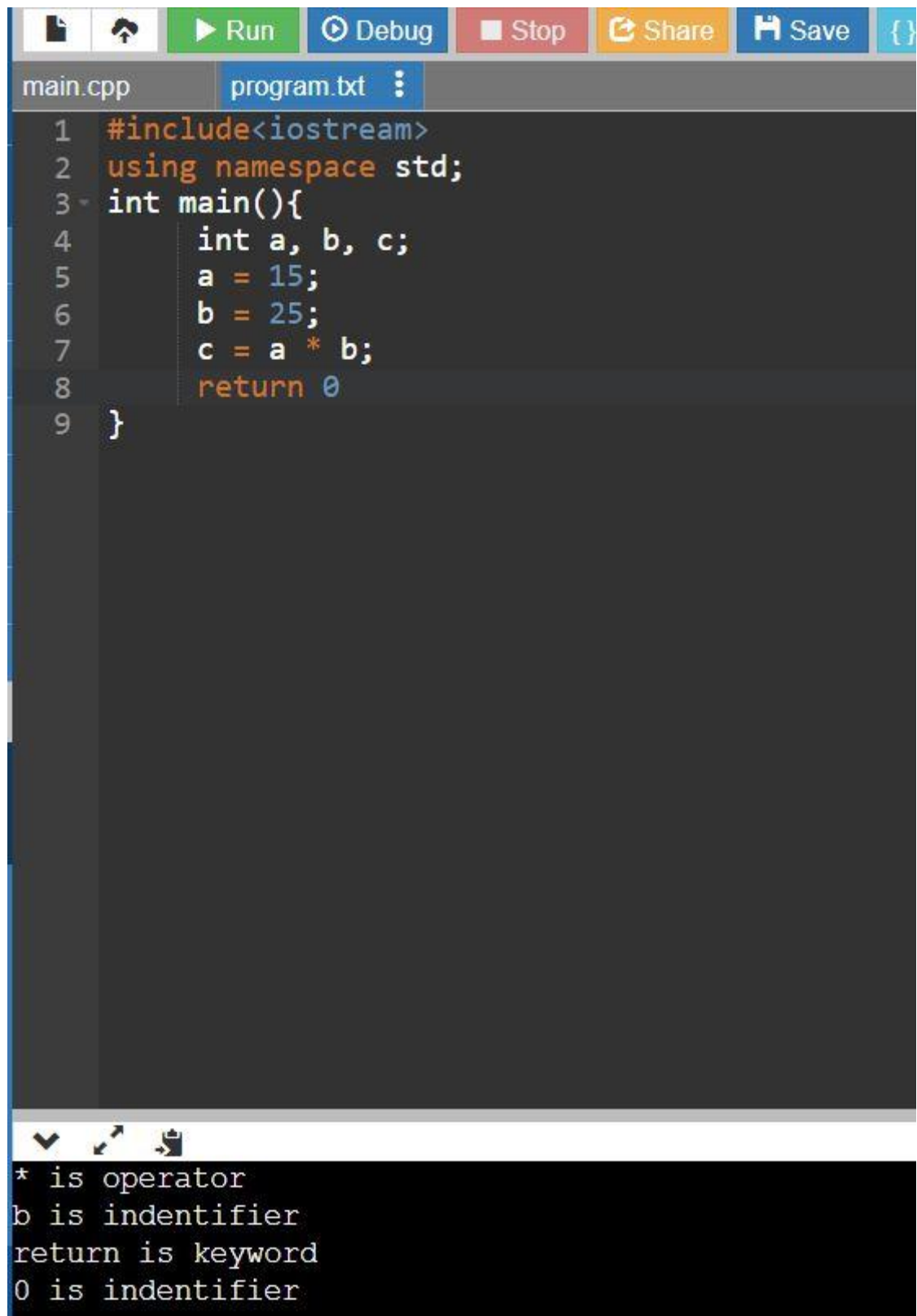
```
#include <iostream>
using namespace std;
int main() {
    int a, b, c;
    a = 15;
    b = 25;
    c = a * b;
    return 0;
}
```

CONSOLE OUTPUT:

* is operator
b is identifier
return is keyword
0 is identifier

The screenshot shows the OnlineGDB website interface. On the left is a sidebar with navigation links: IDE, My Projects, Classroom (new), Learn Programming, Programming Questions, Jobs (new), Sign Up, and Login. Below these are social media icons for Facebook and Twitter, and a red button indicating 86.5K users. A banner for 'GOT AN OPINION? SHARE AND GET REWARDED. ORakuten AIP' is also present. The main area displays a C++ program in a dark-themed editor. The program includes headers for iostream, fstream, stdlib.h, string.h, and ctype.h, and uses the std namespace. It defines a function isKeyword that checks if a character buffer is a keyword from a predefined list. The main function reads characters from 'program.txt' and checks if they are operators, identifiers, keywords, or identifiers. The console output shows the results of these checks for the first few characters: '*' is operator, 'b' is indentifier, 'return' is keyword, and '0' is indentifier. The program finishes with exit code 0.

```
1 #include<iostream>
2 #include<fstream>
3 #include<stdlib.h>
4 #include<string.h>
5 #include<ctype.h>
6
7 using namespace std;
8
9 int isKeyword(char buffer[]){
10 char keywords[32][10] = {"auto","break","case","char","const","continue","default",
11 "do","double","else","enum","extern","float","for","goto",
12 "if","int","long","register","return","short","signed",
13 "sizeof","static","struct","switch","typedef","union",
14 "unsigned","void","volatile","while"};
15 int i, flag = 0;
16 for(i = 0; i < 32; ++i){
17 if(strcmp(keywords[i], buffer) == 0){
18 flag = 1;
19 break;
20 }
21 }
22 return flag;
23 }
24
25 int main(){
26 char ch, buffer[15], operators[] = "+-*/%=";
27 ifstream fin("program.txt");
28
29 * is operator
30 b is indentifier
31 return is keyword
32 0 is indentifier
33
34 ...Program finished with exit code 0
35 Press ENTER to exit console.
```



The image shows a screenshot of a C++ IDE. The top toolbar contains buttons for Run, Debug, Stop, Share, and Save. The file explorer shows two files: main.cpp and program.txt. The main.cpp file is open, displaying the following code:

```
1 #include<iostream>
2 using namespace std;
3 int main(){
4     int a, b, c;
5     a = 15;
6     b = 25;
7     c = a * b;
8     return 0
9 }
```

Below the code editor, there is a tokenization window showing the following tokens:

- * is operator
- b is identifier
- return is keyword
- 0 is identifier

```

emit (0);
}
while (!fin.eof()) {
    ch = fin.get();

    for (i=0; i<6; ++i) {
        if (ch == operators[i])
            cout << ch << " is operator \n";
    }

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

        if (iskeyword(buffer) == 1)
            cout << buffer << " is keyword \n";
        else
            cout << buffer << " is identifier \n";
    }
}

fin.close();
return 0;
}

```

RESULT: Hence the implementation of lexical analyzer in c++ was compiled, executed successfully.