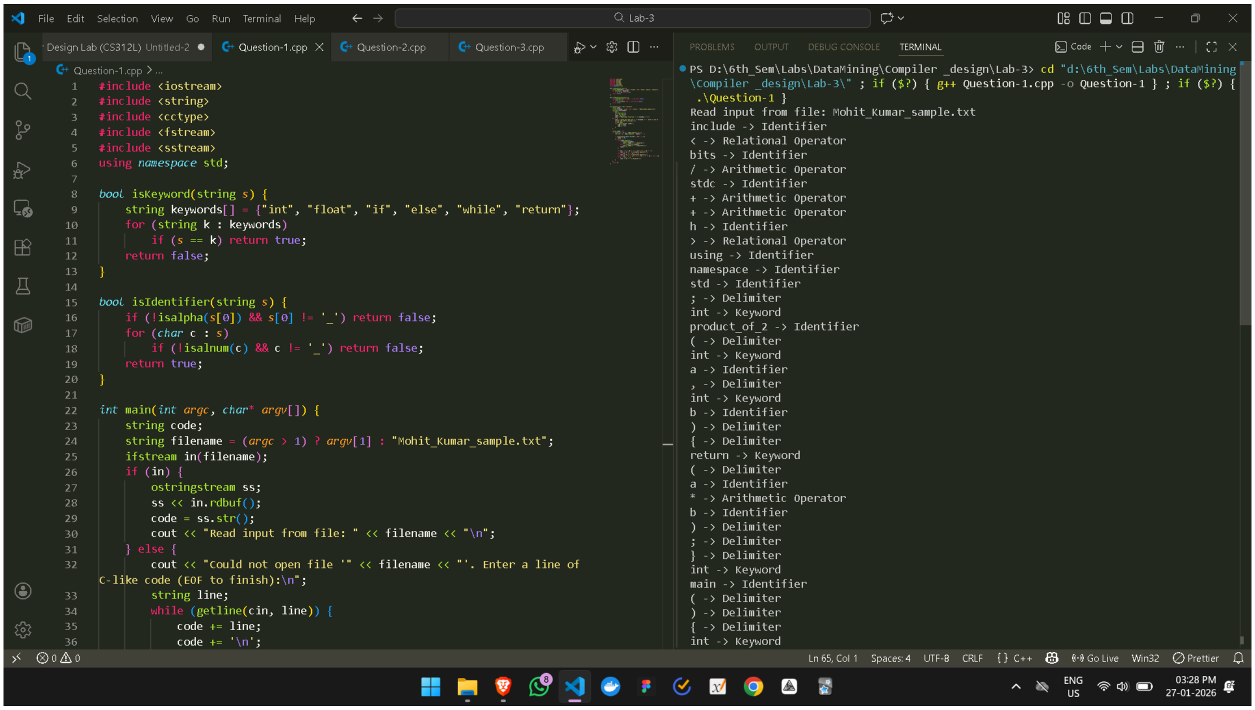
Compiler Design Lab-3

Mohit Kumar 23IT3028

Question-1:
Code:

```cpp
#include <iostream>
#include <string>
#include <cctype>
#include <fstream>
#include <sstream>
using namespace std;

bool isKeyword(string s) {
    string keywords[] = {"int", "float", "if", "else", "while", "return"};
    for (string k : keywords)
        if (s == k) return true;
    return false;
}

bool isIdentifier(string s) {
    if (!isalpha(s[0]) && s[0] != '_') return false;
    for (char c : s)
        if (!isalnum(c) && c != '_') return false;
    return true;
}

int main(int argc, char* argv[]) {
    string code;
    string filename = (argc > 1) ? argv[1] : "Mohit_Kumar_sample.txt";
    ifstream in(filename);
    if (in) {
        ostringstream ss;
        ss << in.rdbuf();
        code = ss.str();
        cout << "Read input from file: " << filename << "\n";
    } else {
        cout << "Could not open file '" << filename << "'. Enter a line of
C-like code (EOF to finish):\n";
        string line;
        while (getline(cin, line)) {
            code += line;
            code += '\n';
        }
    }

    string token = "";
    for (size_t i = 0; i <= code.length(); i++) {
        char ch = (i < code.length()) ? code[i] : ' ';

        if (isalnum((unsigned char)ch) || ch == '_') {
            token += ch;
        } else {
            if (!token.empty()) {
                if (isKeyword(token))
                    cout << token << " -> Keyword\n";
                else if (isIdentifier(token))
                    cout << token << " -> Identifier\n";
                token = "";
            }

            if (ch == '+' || ch == '-' || ch == '*' || ch == '/')
                cout << ch << " -> Arithmetic Operator\n";
            else if (ch == '<' || ch == '>')
                cout << ch << " -> Relational Operator\n";
            else if (ch == ';' || ch == ',' || ch == '(' || ch == ')' ||
ch == '{' || ch == '}')
                cout << ch << " -> Delimiter\n";
        }
    }
    return 0;
}
```

Output:


File Edit Selection View Go Run Terminal Help
Lab-3
Design Lab (CS312L) Untitled-2
Question-1.cpp
Question-2.cpp
Question-3.cpp
Question-1.cpp > ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Code
0 0
Ln 65, Col 1 Spaces: 4 UTF-8 CRLF { } C++ Go Live Win32 Prettier
ENG US
03:28 PM
27-01-2026

```cpp
#include <iostream>
#include <string>
#include <cctype>
#include <fstream>
#include <sstream>
using namespace std;

bool isKeyword(string s) {
    string keywords[] = {"int", "float", "if", "else", "while", "return"};
    for (string k : keywords)
        if (s == k) return true;
    return false;
}

bool isIdentifier(string s) {
    if (!isalpha(s[0]) && s[0] != '_') return false;
    for (char c : s)
        if (!isalnum(c) && c != '_') return false;
    return true;
}

int main(int argc, char* argv[]) {
    string code;
    string filename = (argc > 1) ? argv[1] : "Mohit_Kumar_sample.txt";
    ifstream in(filename);
    if (in) {
        ostringstream ss;
        ss << in.rdbuf();
        code = ss.str();
        cout << "Read input from file: " << filename << "\n";
    } else {
        cout << "Could not open file '" << filename << "'. Enter a line of C-like code (EOF to finish):\n";
        string line;
        while (getline(cin, line)) {
            code += line;
            code += '\n';
```

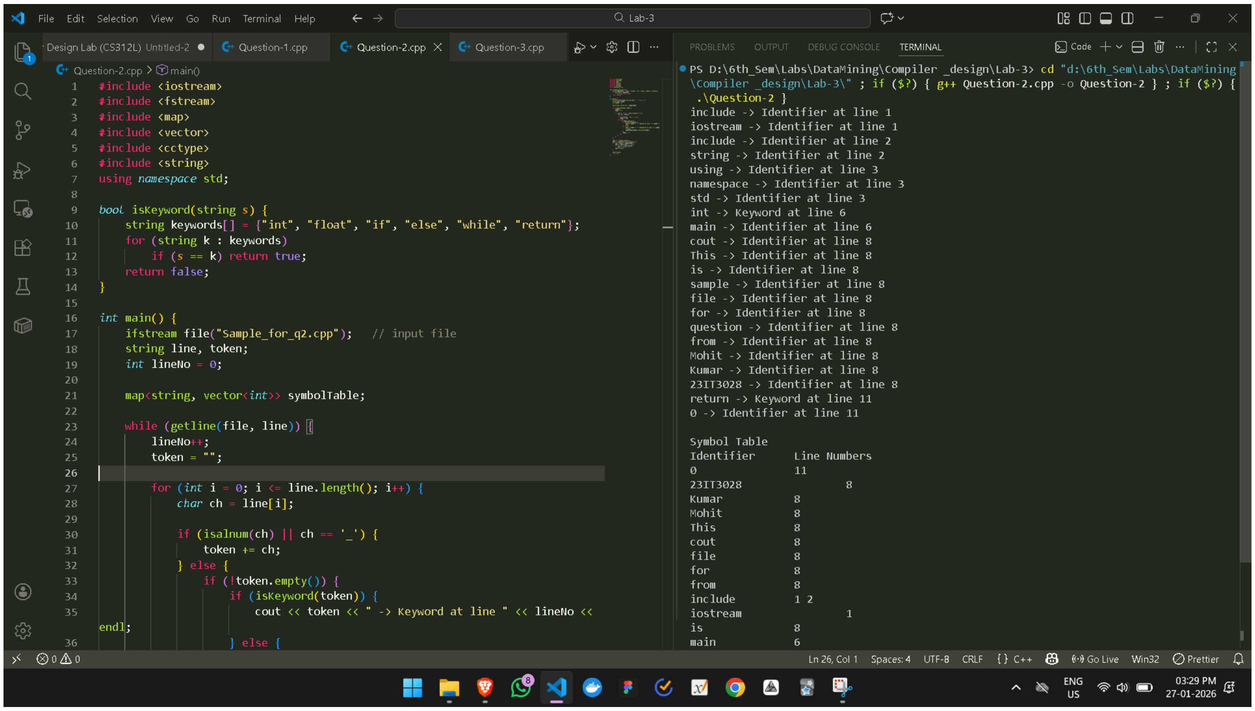
```
PS D:\6th_Sem\Labs\DataMining\Compiler _design\Lab-3> cd "d:\6th_Sem\Labs\DataMining\Compiler _design\Lab-3\" ; if ($?) { g++ Question-1.cpp -o Question-1 } ; if ($?) { .\Question-1 }
Read input from file: Mohit_Kumar_sample.txt
include -> Identifier
< -> Relational Operator
bits -> Identifier
/ -> Arithmetic Operator
stdc -> Identifier
+ -> Arithmetic Operator
+ -> Arithmetic Operator
h -> Identifier
> -> Relational Operator
using -> Identifier
namespace -> Identifier
std -> Identifier
; -> Delimiter
int -> Keyword
product_of_2 -> Identifier
( -> Delimiter
int -> Keyword
a -> Identifier
, -> Delimiter
int -> Keyword
b -> Identifier
) -> Delimiter
{ -> Delimiter
return -> Keyword
( -> Delimiter
a -> Identifier
* -> Arithmetic Operator
b -> Identifier
) -> Delimiter
; -> Delimiter
} -> Delimiter
int -> Keyword
main -> Identifier
( -> Delimiter
) -> Delimiter
{ -> Delimiter
int -> Keyword
```

Question-2:

Code:

```cpp
#include <iostream>
#include <fstream>
#include <map>
#include <vector>
#include <cctype>
#include <string>
using namespace std;

bool isKeyword(string s) {
    string keywords[] = {"int", "float", "if", "else", "while", "return"};
    for (string k : keywords)
        if (s == k) return true;
    return false;
}

int main() {
    ifstream file("Sample_for_q2.cpp");   // input file
    string line, token;
    int lineNo = 0;
    map<string, vector<int>> symbolTable;

    while (getline(file, line)) {
        lineNo++;
        token = "";

        for (int i = 0; i <= line.length(); i++) {
            char ch = line[i];

            if (isalnum(ch) || ch == '_') {
                token += ch;
            } else {
                if (!token.empty()) {
                    if (isKeyword(token)) {
                        cout << token << " -> Keyword at line " << lineNo << endl;
                    } else {
                        cout << token << " -> Identifier at line " << lineNo << endl;
                        symbolTable[token].push_back(lineNo);
                    }
                    token = "";
                }
            }
        }
    }

    cout << "\nSymbol Table\n";
    cout << "Identifier\tLine Numbers\n";
    for (auto &entry : symbolTable) {
        cout << entry.first << "\t\t";
        for (int ln : entry.second)
            cout << ln << " ";
        cout << endl;
    }

    file.close();
    return 0;
}
```

Output:


File Edit Selection View Go Run Terminal Help
Lab-3
Design Lab (CS312L) Untitled-2
Question-1.cpp
Question-2.cpp
Question-3.cpp
Question-2.cpp > main()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Code
Ln 26, Col 1 Spaces: 4 UTF-8 CRLF { } C++ Go Live Win32 Prettier
ENG US
03:29 PM
27-01-2026

```cpp
#include <iostream>
#include <fstream>
#include <map>
#include <vector>
#include <cctype>
#include <string>
using namespace std;

bool isKeyword(string s) {
    string keywords[] = {"int", "float", "if", "else", "while", "return"};
    for (string k : keywords)
        if (s == k) return true;
    return false;
}

int main() {
    ifstream file("Sample_for_q2.cpp");   // input file
    string line, token;
    int lineNo = 0;

    map<string, vector<int>> symbolTable;

    while (getline(file, line)) {
        lineNo++;
        token = "";

        for (int i = 0; i <= line.length(); i++) {
            char ch = line[i];

            if (isalnum(ch) || ch == '_') {
                token += ch;
            } else {
                if (!token.empty()) {
                    if (isKeyword(token)) {
                        cout << token << " -> Keyword at line " << lineNo <<
endl;
                    } else {
```

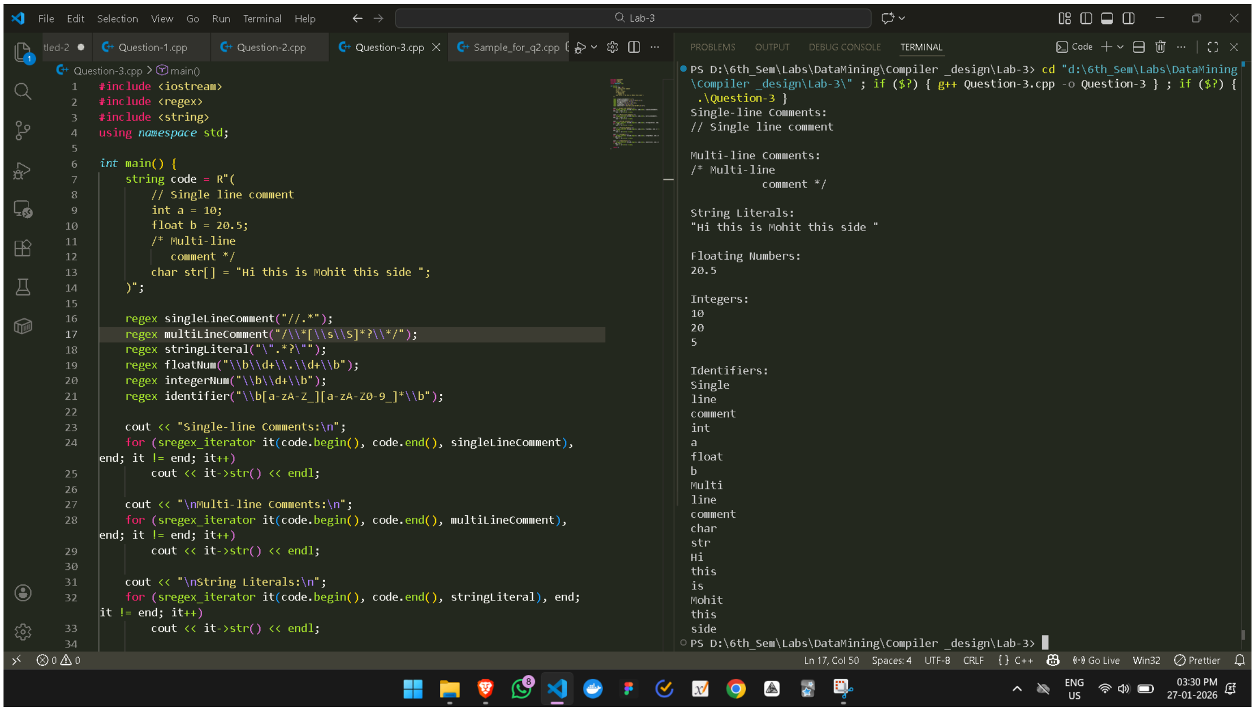
```
PS D:\6th_Sem\Labs\DataMining\Compiler _design\Lab-3> cd "d:\6th_Sem\Labs\DataMining
\Compiler _design\Lab-3\" ; if ($?) { g++ Question-2.cpp -o Question-2 } ; if ($?) {
 .\Question-2 }
include -> Identifier at line 1
iostream -> Identifier at line 1
include -> Identifier at line 2
string -> Identifier at line 2
using -> Identifier at line 3
namespace -> Identifier at line 3
std -> Identifier at line 3
int -> Keyword at line 6
main -> Identifier at line 6
cout -> Identifier at line 8
This -> Identifier at line 8
is -> Identifier at line 8
sample -> Identifier at line 8
file -> Identifier at line 8
for -> Identifier at line 8
question -> Identifier at line 8
from -> Identifier at line 8
Mohit -> Identifier at line 8
Kumar -> Identifier at line 8
23IT3028 -> Identifier at line 8
return -> Keyword at line 11
0 -> Identifier at line 11

Symbol Table
Identifier      Line Numbers
0               11
23IT3028                8
Kumar           8
Mohit           8
This            8
cout            8
file            8
for             8
from            8
include         1 2
iostream                1
is              8
main            6
```

Question-3:

Code:

```cpp
#include <iostream>
#include <regex>
#include <string>
using namespace std;

int main() {
    string code = R"(
        // Single line comment
        int a = 10;
        float b = 20.5;
        /* Multi-line
           comment */
        char str[] = "Hello World";
    )";

    regex singleLineComment("//.*");
    regex multiLineComment("/\\*[\\s\\S]*?\\*/");
    regex stringLiteral("\".*?\"");
    regex floatNum("\\b\\d+\\.\\d+\\b");
    regex integerNum("\\b\\d+\\b");
    regex identifier("\\b[a-zA-Z_][a-zA-Z0-9_]*\\b");

    cout << "Single-line Comments:\n";
    for (sregex_iterator it(code.begin(), code.end(), singleLineComment),
end; it != end; it++)
        cout << it->str() << endl;

    cout << "\nMulti-line Comments:\n";
    for (sregex_iterator it(code.begin(), code.end(), multiLineComment),
end; it != end; it++)
        cout << it->str() << endl;

    cout << "\nString Literals:\n";
    for (sregex_iterator it(code.begin(), code.end(), stringLiteral), end;
it != end; it++)
        cout << it->str() << endl;

    cout << "\nFloating Numbers:\n";
    for (sregex_iterator it(code.begin(), code.end(), floatNum), end; it
!= end; it++)
        cout << it->str() << endl;

    cout << "\nIntegers:\n";
    for (sregex_iterator it(code.begin(), code.end(), integerNum), end; it
!= end; it++)
        cout << it->str() << endl;

    cout << "\nIdentifiers:\n";
    for (sregex_iterator it(code.begin(), code.end(), identifier), end; it
!= end; it++)
        cout << it->str() << endl;

    return 0;
}
```

Output:



```cpp
#include <iostream>
#include <regex>
#include <string>
using namespace std;

int main() {
    string code = R"(
        // Single line comment
        int a = 10;
        float b = 20.5;
        /* Multi-line
           comment */
        char str[] = "Hi this is Mohit this side ";
    )";

    regex singleLineComment("//.*");
    regex multiLineComment("/\\*[\\s\\S]*?\\*/");
    regex stringLiteral("\".*?\"");
    regex floatNum("\\b\\d+\\.\\d+\\b");
    regex integerNum("\\b\\d+\\b");
    regex identifier("\\b[a-zA-Z_][a-zA-Z0-9_]*\\b");

    cout << "Single-line Comments:\n";
    for (sregex_iterator it(code.begin(), code.end(), singleLineComment),
end; it != end; it++)
        cout << it->str() << endl;

    cout << "\nMulti-line Comments:\n";
    for (sregex_iterator it(code.begin(), code.end(), multiLineComment),
end; it != end; it++)
        cout << it->str() << endl;

    cout << "\nString Literals:\n";
    for (sregex_iterator it(code.begin(), code.end(), stringLiteral), end;
it != end; it++)
        cout << it->str() << endl;

```

```
PS D:\6th_Sem\Labs\DataMining\Compiler _design\Lab-3> cd "d:\6th_Sem\Labs\DataMining
\Compiler _design\Lab-3\" ; if ($?) { g++ Question-3.cpp -o Question-3 } ; if ($?) {
 .\Question-3 }
Single-line Comments:
// Single line comment

Multi-line Comments:
/* Multi-line
           comment */

String Literals:
"Hi this is Mohit this side "

Floating Numbers:
20.5

Integers:
10
20
5

Identifiers:
Single
line
comment
int
a
float
b
Multi
line
comment
char
str
Hi
this
is
Mohit
this
side
PS D:\6th_Sem\Labs\DataMining\Compiler _design\Lab-3>
```