INTERNATIONAL ISLAMIC UNIVERSITY CHITTAGONG

Department Of Computer Science and Engineering

MID LAB ASSIGNMENT

Name: Nowrin Akter Mahi

Id: C221251

Section: 5BF

Course-code: CSE-3528

Course-title: Compiler Lab

Submitted-to: Jafrin Iqbal Chowdhury

Adjunct Faculty of CSE at IIUC

1. VALID IDENTIFIER:

C++ code:

```
#include<iostream>

using namespace std;
bool isValid(string str)
{
   int n=str.size();
   if((str[0]>='a'&&str[0]<='z')||(str[0]>='A'&&str[0]<='Z')||(str[0]=='_'))
   {
      for(int i=1;i<n;i++)
      {
         if((str[i]>='a'&&str[i]<='z')||(str[i]>='A'&&str[i]<='Z')||(str[i]=='_')||(str[i]>='0'&&str[i]<='9'));
         else
         {
            return false;
         }
      }
   }
   else
   {
      return false;
   }
   //return true;
}
int main()
{
   string str;
   getline(cin,str);
   bool flag=isValid(str);
{
   if(flag==true)
   {
      cout<<"valid identifier"<<endl;
   }
   else
   {
      cout<<"Invalid identifier"<<endl;
   }
}
   return 0;
}
```

Lex code:

```
%{
#include <iostream>
using namespace std;
%}

/* Definitions*/
letter [a-zA-Z]
digit [0-9]
underscore _

/* Rules*/
%%
{letter}({letter}|{digit}|{underscore})* {
   cout << "Valid identifier" << endl;
}

.|\n {
   /* Ignore other characters*/
}

%%
int main() {
   yylex();
   return 0;
}
```

2. TOKERIZATION:

C++ code:

```
#include<stdio.h>
int main()
{
    int a,b,c;
    printf("enter two number: ");
    scanf("%d%d",&a,&b);
    c=a+b;
    printf("%d",c);
    return 0;
}
```

Lex code:

```
%{
#include <stdio.h>
%}

%%

[ \t\n] ; /* Ignore whitespace */

[0-9]+ {
   printf("NUMBER\n");
}

"enter"|"two"|"number" {
   printf("KEYWORD\n");
}

"printf"|"scanf" {
   printf("IO_FUNCTION\n");
}

"+"|"-"|"*"|"/" {
   printf("OPERATOR\n");
}

"(" {
   printf("LEFT_PAREN\n");
}

")" {
   printf("RIGHT_PAREN\n");
}

";" {
   printf("STATEMENT_END\n");
}
%%
int main() {
   yylex();
   return 0;
}
```

3.Single Comment and Multiple Comment Remove:

C++ code:

```
#include <iostream>
#include <string>

using namespace std;

int main() {
   string line;
   bool inSingleLineComment = false;
   bool inMultiLineComment = false;

   while (getline(cin, line)) {
      string result;
      for (size_t i = 0; i < line.length(); ++i) {
         if (!inSingleLineComment && !inMultiLineComment && line[i] == '/' && i + 1 <
line.length() && line[i + 1] == '/') {
            // Start of single-line comment
            break; // Skip the rest of the line
         } else if (!inSingleLineComment && !inMultiLineComment && line[i] == '/' && i + 1 <
line.length() && line[i + 1] == '*') {
            // Start of multi-line comment
            inMultiLineComment = true;
            ++i; // Skip the next character '*'
         } else if (inMultiLineComment && line[i] == '*' && i + 1 < line.length() && line[i + 1] == '/')
{
            // End of multi-line comment
            inMultiLineComment = false;
            ++i; // Skip the next character '/'
         } else if (!inSingleLineComment && !inMultiLineComment) {
            // Not in comment, add character to result
            result += line[i];
         }
      }
      // Output the result without comments
      cout << result << endl;
   }
   return 0;
```

```
}
```

Lex code:

```
%{
#include <iostream>
using namespace std;
%}

%option noyywrap

%%
"//".*     /* Remove single-line comments */
"/\*"      { /* Start of multi-line comment */
           while (1) {
              char c = yyinput();
              if (c == '*') {
                 char next = yyinput();
                 if (next == '/') break; // End of multi-line comment
                 unput(next);
              }
           }
         }
.          /* Default rule: echo all other characters */
%%

int yyinput() {
   return cin.get();
}

int main() {
   yylex();
   return 0;
}
```