

```
#ifndef mcrlex_h
#define mcrlex_h 1
//-----
// File mcrlex.h defines class Lexer.
//-----
// Author: Thomas R. Turner
// E-Mail: trturner.uco.edu
// Date: March, 2003
//-----
// Revised January, 2015
//-----
// Copyright March, 2003 by Thomas R. Turner
// Do not reproduce without permission from Thomas R. Turner.
//-----
//-----
// Standard C and C++ include files
//-----
#include <stdio>
#include <fstream>
#include <iostream>
using namespace std;
//-----
//Token definitions
//-----
#include "y.tab.h"
//-----
//Function: yylex
//Function yylex is the mcrner. Function yylex returns an integer
//token code as defined above or 0 if end-of-file has been
//reached.
//-----
#ifdef __cplusplus
extern "C"
#endif
int yylex (void);
//-----
//Class Lexer defines the attributes of a Scanner
//-----
class Lexer {
    int tokencode;          //Code for the most recent token found
public:
    Lexer(FILE* i);         //Constructor used to redirect the keyboard
                           //(stdin) to file i.
    int Lex(void);          //Call the scanner yylex and return the code
                           //found by yylex
    int FetchTokenCode(void); //Return the code of the most recent token
    void StoreTokenCode(int T); //Store the token code.
```

```
char* FetchSpelling(void); //Return the spelling of the most recent
                           //token
};
#endif
```