

```

1  %{
2  //Se incluyen los header correspondientes para poder usar ciertas funciones y
  reconocer los TOKEN.
3  #include <string.h>
4  #include "y.tab.h"
5  %{
6
7
8  %{
9      /*
10         Lo siguiente son las declaraciones léxicas de Micro.
11
12         %% separa las definiciones de LEX.
13         %{ %} se utiliza para escribir código de C que será copiado literalmente.
14     */
15     %{
16
17     PALABRA_RESERVADA (inicio|fin|leer|escribir)
18     LETRA [a-zA-Z]
19     DIGITO [0-9]
20     OPERADOR_ADITIVO      [+ -]
21     ASIGNACIÓN :=
22
23     %%
24
25     [ ( , ; ]          return yytext[0];
26     inicio             return INICIO;
27     fin                return FIN;
28     leer               return LEER;
29     escribir           return ESCRIBIR;
30
31     {LETRA} ({LETRA}|{DIGITO})* {
32                                     //Almacenamos el nombre del identificador
                                     reconocido para luego poder acceder al mismo
                                     desde YACC.
33                                     //Si no se utiliza strdup() el puntero de yytext
                                     será copiado en yylval.identificador.
34                                     yylval.identificador = strdup(yytext);
35                                     printf("Identificador: %s\n", yytext);
36                                     return IDENTIFICADOR; //Retornamos el token
                                     IDENTIFICADOR para YACC.
37                                     }
38     {DIGITO}{DIGITO}* {
39                                     printf("Constante: %s\n", yytext);
40                                     return CONSTANTE;
41                                     }
42     {OPERADOR_ADITIVO} {
43                                     printf("Operador aditivo: %s\n", yytext);
44                                     return OPERADOR_ADITIVO;
45                                     }
46     {ASIGNACION} {
47                                     printf("Asignacion: %s\n", yytext);
48                                     return ASIGNACION;
49                                     }
50     [ \t\n]+          /* ignorar espacios y enter, no nos importa
51     analizarlos. */;
52
53
54
55
56

```