

```

E                [Ee]
OtherLetter      [A-DF-Za-df-z]
Letter           {E} | {OtherLetter}
IntLit           {Digit}+
%%
[ \t\n]+         { /* delete */           }
[Bb][Ee][Gg][Ii][Nn] { minor=0; return(4); }
[Ee][Nn][Dd]      { minor=0; return(5); }
[Rr][Ee][Aa][Dd]  { minor=0; return(6); }
[Ww][Rr][Ii][Tt][Ee] { minor=0; return(7); }
{Letter}({Letter} | {Digit} | _)* { minor=0; return(1); }
{IntLit}          { minor=1; return(2); }
({IntLit}[.]{IntLit})(E)[+-]?{IntLit})? { minor=2; return(2); }
\"([^\n\" | \"\")*\" { stripquotes();
                        minor=3; return(2); }
\"([^\n\" | \"\")*\" { stripquotes();
                        minor=0; return(3); }
“(” { minor=0; return(8); }
”)” { minor=0; return(9); }
“,” { minor=0; return(10); }
“,” { minor=0; return(11); }
“:=” { minor=0; return(12); }
“+” { minor=0; return(13); }
“-” { minor=0; return(14); }
%%
/* Strip unwanted quotes from string in yytext;  adjust yyleng. */
void stripquotes(void)
{
    Int frompos, topos = 0, numquotes = 2;

    for (frompos = 1; frompos < yyleng ; frompos++) {
        yytext[topos++] = yytext[frompos];
        if (yytext[frompos] == “” && yytext[frompos+1] == “”) {
            frompos++;
            numquotes++;
        }
    }
    yyleng -= numquotes;
    yytext[yyleng] = ‘\0’;
}

```

Figure 3.5 A Lex Definition for Extended Micro