```
%{
    #include "y.tab.h"
    int countn=0;
%}
%option yylineno

alpha [a-zA-Z]
digit [0-9]
unary "++"|"--"

%%

"printf"                    { strcpy(yylval.nd_obj.name,(yytext)); return PRINTFF; }
"scanf"                     { strcpy(yylval.nd_obj.name,(yytext)); return SCANFF; }
"int"                       { strcpy(yylval.nd_obj.name,(yytext)); return INT; }
"float"                     { strcpy(yylval.nd_obj.name,(yytext)); return FLOAT; }
"char"                      { strcpy(yylval.nd_obj.name,(yytext)); return CHAR; }
"void"                      { strcpy(yylval.nd_obj.name,(yytext)); return VOID; }
"return"                    { strcpy(yylval.nd_obj.name,(yytext)); return RETURN; }
"for"                       { strcpy(yylval.nd_obj.name,(yytext)); return FOR; }
"if"                        { strcpy(yylval.nd_obj.name,(yytext)); return IF; }
"else"                      { strcpy(yylval.nd_obj.name,(yytext)); return ELSE; }
^"#include"[ ]*<.+\.h>      { strcpy(yylval.nd_obj.name,(yytext)); return INCLUDE; }
"true"                      { strcpy(yylval.nd_obj.name,(yytext)); return TRUE; }
"false"                     { strcpy(yylval.nd_obj.name,(yytext)); return FALSE; }
[-]?{digit}+                { strcpy(yylval.nd_obj.name,(yytext)); return NUMBER; }
[-]?{digit}+\.{digit}{1,6}  { strcpy(yylval.nd_obj.name,(yytext)); return FLOAT_NUM; }
{alpha}({alpha}|{digit})*   { strcpy(yylval.nd_obj.name,(yytext)); return ID; }
{unary}                     { strcpy(yylval.nd_obj.name,(yytext)); return UNARY; }
"<="                        { strcpy(yylval.nd_obj.name,(yytext)); return LE; }
">="                        { strcpy(yylval.nd_obj.name,(yytext)); return GE; }
"=="                        { strcpy(yylval.nd_obj.name,(yytext)); return EQ; }
"!="                        { strcpy(yylval.nd_obj.name,(yytext)); return NE; }
">"                             { strcpy(yylval.nd_obj.name,(yytext)); return GT; }
"<"                             { strcpy(yylval.nd_obj.name,(yytext)); return LT; }
"&&"                        { strcpy(yylval.nd_obj.name,(yytext)); return AND; }
"||"                        { strcpy(yylval.nd_obj.name,(yytext)); return OR; }
"+"                         { strcpy(yylval.nd_obj.name,(yytext)); return ADD; }
"-"                         { strcpy(yylval.nd_obj.name,(yytext)); return SUBTRACT; }
"/"                         { strcpy(yylval.nd_obj.name,(yytext)); return DIVIDE; }
"*"                         { strcpy(yylval.nd_obj.name,(yytext)); return MULTIPLY; }
\/\/.*                      { ; }
\/\*(.*\n)*.*\*\/           { ; }
[ \t]*                      { ; }
[\n]                        { countn++; }
.                               { return *yytext; }
["].*["]                    { strcpy(yylval.nd_obj.name,(yytext)); return STR; }
['].[']                     { strcpy(yylval.nd_obj.name,(yytext)); return CHARACTER; }

%%

int yywrap() {
    return 1;
}
```