Commands

flex spec.lxi

gcc lex.yy.c -o result

result.exe p1.txt

Spec.lxi

```
%{
#include <math.h>
int lines = 0;
%}
%option noyywrap

DIGIT		[0-9]
NUMBER		[1-9][O-9]*
STRING		\"[a-zA-Z]*\"
CONSTANT	{STRING}|{DIGIT}
ID		[a-zA-Z][a-zA-Z0-9]*

%%

"programa"		{printf( "Reserved word: %s\n", yytext ); }
"finprograma"		{printf( "Reserved word: %s\n", yytext ); }
"listo"		{printf( "Reserved word: %s\n", yytext ); }
"leer"		{printf( "Reserved word: %s\n", yytext ); }
"variable"		{printf( "Reserved word: %s\n", yytext ); }
"entero"		{printf( "Reserved word: %s\n", yytext ); }
"doble"		{printf( "Reserved word: %s\n", yytext ); }
"booleano"		{printf( "Reserved word: %s\n", yytext ); }
"char"		{printf( "Reserved word: %s\n", yytext ); }
```

```
"fi"            {printf( "Reserved word: %s\n", yytext ); }
"filse"         {printf( "Reserved word: %s\n", yytext );}
"lse"           {printf( "Reserved word: %s\n", yytext ); }
"fiend"         {printf( "Reserved word: %s\n", yytext );}
"entonces"              {printf( "Reserved word: %s\n", yytext ); }
"iniciobucle"           {printf( "Reserved word: %s\n", yytext ); }
"finbucle"              {printf( "Reserved word: %s\n", yytext ); }
"desde"                 {printf( "Reserved word: %s\n", yytext ); }
"hasta"                 {printf( "Reserved word: %s\n", yytext ); }
"imprime"               {printf( "Reserved word: %s\n", yytext ); }
"devolver"              {printf( "Reserved word: %s\n", yytext ); }
"romper"                {printf( "Reserved word: %s\n", yytext ); }
"and"           {printf( "Operator: %s\n", yytext ); }
"or"            {printf( "Operator: %s\n", yytext ); }

{ID}            {printf( "Identifier: %s\n", yytext ); }

{CONSTANT}              {printf( "Constant: %s\n", yytext ); }

"+"             {printf( "Operator: %s\n", yytext ); }
"-"             {printf( "Operator: %s\n", yytext ); }
"*"             {printf( "Operator: %s\n", yytext ); }
"/"             {printf( "Operator: %s\n", yytext ); }
"%"             {printf( "Operator: %s\n", yytext ); }
"<="            {printf( "Operator: %s\n", yytext ); }
"<"             {printf( "Operator: %s\n", yytext ); }
"=="            {printf( "Operator: %s\n", yytext ); }
">="            {printf( "Operator: %s\n", yytext ); }
">"             {printf( "Operator: %s\n", yytext ); }
```

```
"="             {printf( "Operator: %s\n", yytext ); }
"+="            {printf( "Operator: %s\n", yytext ); }
"-="            {printf( "Operator: %s\n", yytext ); }

"("             {printf( "Separator: %s\n", yytext ); }
")"             {printf( "Separator: %s\n", yytext ); }
"["             {printf( "Separator: %s\n", yytext ); }
"]"             {printf( "Separator: %s\n", yytext ); }
"{"             {printf( "Separator: %s\n", yytext ); }
"}"             {printf( "Separator: %s\n", yytext ); }
":"             {printf( "Separator: %s\n", yytext ); }
";"             {printf( "Separator: %s\n", yytext ); }

[ \t]+
[\n]+           {++lines;}

.               {printf( "Illegal symbol at line %d\n", lines); return -1;}


%%
main( argc, argv )
int argc;
char **argv;
{
   ++argv, --argc; /* skip over program name */
   if ( argc > 0 )
   yyin = fopen( argv[0], "r" );
   else
    yyin = stdin;
```

```
    yylex();
}
```

P1.txt

```
programa
leer entero a;
leer entero b;
variable entero resultado;
fi (a>=b) entonces resultado=a fiend;
filse (b>=a) entonces resultado=b fiend;
imprime resultado;
finprograma
```

Demo output

```
Reserved word: programa
Reserved word: leer
Reserved word: entero
Identifier: a
Separator: ;
Reserved word: leer
Reserved word: entero
Identifier: b
Separator: ;
Reserved word: variable
Reserved word: entero
Identifier: resultado
Separator: ;
Reserved word: fi
Separator: (
```

Identifier: a

Operator: >=

Identifier: b

Separator:)

Reserved word: entonces

Identifier: resultado

Operator: =

Identifier: a

Reserved word: fiend

Separator: ;

Reserved word: filse

Separator: (

Identifier: b

Operator: >=

Identifier: a

Separator:)

Reserved word: entonces

Identifier: resultado

Operator: =

Identifier: b

Reserved word: fiend

Separator: ;

Reserved word: imprime

Identifier: resultado

Separator: ;

Reserved word: finprograma