LEX lab

Github url:

https://github.com/Akitektuo/University/tree/master/3rd%20year/FLCT/lab/Lab9

compile.sh

```
flex specif.lxi
gcc lex.yy.c -o exe -ll
./exe < inputFiles/p1.txt > outputFiles/output.txt
cat outputFiles/output.txt
```

specif.lxi

```
%{
#include <stdio.h>
#include <string.h>
int lines = 0;
%}

%option noyywrap
%option caseless
RESERVED
\b("int"|bool|string|input|print|when|otherwise|in|while|each)\b
BOOL            \b(false|true)\b
STRING          \"(\\\"|[^\"])*\"
NUMBER          \b(0|([+\-]?[1-9]\d*))\b
CONST           {NUMBER}|{BOOL}|{STRING}
ID                      [a-zA-Z][0-9a-zA-Z_]*
OPERATOR
"=="|"="|"\+\+"|"\+="|"\+"|"--"|"-="|"-"|"\*="|"\*"|"/="|"/"|"%="|"%"
|"!="|"<="|"<"|">="|">"|"!"|"&="|"\|="|"&"|"\|"
SEPARATOR       [()}{\]\[ "]

%%

{RESERVED}      {printf( "Reserved: %s\n", yytext); }

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

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

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

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

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

%%
```