

# Laboratorio di Compilatori

Michele Peroli



REsearch Group in Information Security  
Department of Computer Science  
University of Verona, Italy

2 MAGGIO 2013

Per ciascuno dei seguenti problemi, scrivere un programma Lex che ne realizza una soluzione.

- Sostituire tutte le vocali sia minuscole che maiuscole che compaiono in un testo di input con occorrenze di vocali diverse da quelle lette.
- Contare il numero delle righe e dei caratteri in un testo.
- Invertire le lettere che compongono una data stringa in input.
- Stampare a schermo la stringa letta.

- Scrivere un riconoscitore per rispondere alle parole inserite.
- Realizzare uno scanner per un linguaggio simil-Pascal che contiene
  - identificatori, i.e. sequenze di lettere minuscole e cifre che cominciano con una lettera;
  - numeri interi e float;
  - parole chiave: if, then, begin, end, procedure, function;
  - operatori: +, -, \*, /

Contare il numero delle righe e dei caratteri in un testo.

```
int n_linee = 0, n_caratteri = 0;

%option noyywrap
%%
\n {n_linee++; n_caratteri++;}
. {n_caratteri++;}
"esci" {return (1);}
%%
main () {
yylex ();
printf ("linee:%d  caratteri:%d\n",n_linee, n_caratteri);
}
```

Invertire le lettere che compongono una data stringa in input.

```
%option main
%%
[a-zA-Z ]+ { int i;
for (i = 1; i <= yyleng; i++)
printf ("%c", yytext[yyvaleng-i]); }
%%
```

Sostituzione delle vocali in una stringa.

```
%option noyywrap
%%
[aA]+ printf("e");
[eE]+ printf("i");
[iI]+ printf("o");
[oO]+ printf("u");
[uU]+ printf("a");
.;
%%
```

Sostituzione delle vocali in una stringa.

```
main (argc, argv)
int argc; char **argv;
{
++argv, --argc; /* cos ignoriamo il nome
del programma lanciato */
if (argc > 0) {yyin = fopen (argv[0], "r");}
else {printf ("Leggo da tastiera!\n"); yyin = stdin;}
yylex ();
}
```

Scrivere un riconoscitore per rispondere alle parole inserite.

```
%option main
%%
"antoni[o]+"      ECHO;
"giampier[o]+" {printf("Che ti rode?");}
"banana" |
"maleducato" {printf("%s sarai tu!", yytext);}
[A-Z ]+ {printf("%s? Hmmm...");}
[a-z ]+  ECHO;
%%
```

## Realizzare uno scanner per un linguaggio simil-Pascal

```
/* scanner per un linguaggio simil-Pascal */
%{
#include <math.h>
%}
DIGIT    [0-9]
ID       [a-z][a-z0-9]*
%option noyywrap
%%
{DIGIT}+ { printf( "Numero intero: %s (%d)\n",
                  yytext, atoi( yytext ) );
          }
```

## Realizzare uno scanner per un linguaggio simil-Pascal

```
{DIGIT}+"."{DIGIT}*      {  
    printf( "Numero reale: %s (%g)\n",  
           yytext, atof( yytext ) );  
}
```

```
if|then|begin|end|procedure|function  
    {  
    printf( "Parola chiave: %s\n", yytext );  
    }
```

## Realizzare uno scanner per un linguaggio simil-Pascal

```
{ID}          printf( "Identificatore: %s\n", yytext );
"+"|"-"|"*"|"/" printf("identificatore: %s\n", yytext);
"{ "[^}\n]*" }
[ \t\n]+
.             printf( "Carattere sconosciuto: %s\n",
                    yytext );
%%
```

## Realizzare uno scanner per un linguaggio simil-Pascal

```
main( argc, argv )
int argc; char **argv;
    { ++argv, --argc;
    if ( argc > 0 ) yyin = fopen( argv[0], "r" );
    else yyin = stdin;
    yylex();
}
```

Realizzare un programma in flex che somma due numeri romani