

Corso Introduzione ai compilatori

Università di Napoli

Proposta di progetto d'esame

Realizzare un traduttore tra due linguaggi di programmazione, uno sorgente (da tradurre) e l'altro obiettivo (tradotto); per esempio: C - Pascal, ASP-JSP. Bisogna indicare con chiarezza eventuali limitazioni adottate, e includere in commenti del linguaggio obbiettivo i costrutti del linguaggio sorgente non traducibili.

Il traduttore si può realizzare utilizzando Lex e YACC.

L'analizzatore lessicale deve essere in grado di riconoscere i token del linguaggio sorgente. Se si accedere direttamente alle stringhe del linguaggio sorgente nelle produzioni YACC, bisogna ridefinire il tipo del token corrispondente. Es, nel caso degli identificatori:

In YACC:

```
%union {
    char* str_value;
}

%type <str_value> IDENTIFIER
...
%%
...
expression
    : VARIABLE      { printf("%s", make_var($1, "int") ); free($1); }
    | INTEGER        { printf("%s", $1); free($1); }
    | FLOAT          { printf("%s", $1); free($1); }
    | STRING         { printf("%s", $1); free($1); }
    | IDENTIFIER     { printf("%s", $1); free($1); }
```

In Lex:

```
identifier  [A-Za-z_][A-Za-z0-9_]*
...
%%
...
{identifier}      { yylval.str_value = new_string(yytext, yyleng);
                   return(IDENTIFIER); }
...
%%
...
char *new_string(char *s, int n) {
    char *t = malloc(n+1);
    strncpy(t, s, n);
    t[n]='\0';
    return t;
}
```

Possono essere utilizzate produzioni embedded per tradurre “simultaneamente” alcuni costrutti, come ad esempio:

```
block
    : '{' { printf("{"); } statement_list '}' { printf("}"); }
    ;
```