CD LAB 6 Semantic Analyzer using YACC

Bhushan Sonawane
BE E 66 (BATCH 3)

yacc.y

```
%{
#include <stdio.h>
#include <malloc.h>
#include <string.h>
char vartype[10];
struct STable{
char label[10];
char type[10];
int size;
int location;
struct STable* next;
}*head;

void yyerror(const char *st){}
%}

%union { struct sym{ char* label; char type[10]; }SM; }
%token NL
%token <SM> ID
%token  INT FLOAT CHAR DOUBLE
%type <SM> E
%%

S: S Declare
   | Declare
   | S Assign
   | Assign
    ;

Declare: Type List ';'
    ;

Type: INT {strcpy(vartype ,"INT"); }
    | FLOAT {strcpy(vartype ,"FLOAT"); }
    | CHAR {strcpy(vartype ,"CHAR");}
    | DOUBLE {strcpy(vartype ,"DOUBLE");}
    ;

List : List ',' ID { newSYM($3.label,vartype);}
    | ID { newSYM($1.label,vartype);}
    ;

Assign: E '=' E ';' {
                        if(strcmp($1.type,$3.type) != 0){
                          printf("\nError: Type Mismatch %s and %s",$1.type,$3.type);
                         }
```

```
                }

        ;

E: E '+' E          {
                     if(strcmp($1.type,$3.type) != 0){
                         printf("\nError: Type Mismatch %s and %s",$1.type,$3.type);
                     }
                  }
    |
    E '-' E         {
                    if(strcmp($1.type,$3.type) != 0){
                         printf("\nError: Type Mismatch %s and %s",$1.type,$3.type);
                     }
                  }
    | ID            {
                     if(!isDeclared($1.label)){
                                printf("\nError: Undeclared Variable %s",$1.label);
                     }else{
                         strcpy($$.type,getType($1.label));
                     }
                  }
  ;

%%
void DisplaySTable(struct STable*);
int isDeclared(char* lab){
    struct STable *st = head;
    while(st){
     if(strcmp(st->label,lab) == 0){
        return 1;
     }
     st = st->next;
}
return 0;
}
char* getType(char* lab){
    struct STable *st = head;
    while(st){
     if(strcmp(st->label,lab) == 0){
        return st->type;
     }
     st = st->next;
}
}

int main(){
stdin = fopen("in","r");
freopen("out","w",stdout);
```

CD LAB 6 Semantic Analyzer using YACC

```
yyparse();
DisplaySTable(head);
}

int getVarSize(char* st){
if(strcmp(st,"INT") == 0 ||strcmp(st,"FLOAT") == 0  )
   return 4;
if(strcmp(st,"CHAR") == 0 )
   return 1;
if(strcmp(st,"DOUBLE") == 0  )
   return 8;
}

void newSYM(char* lab, char* vartype){
struct STable *tnode = head;
int size = getVarSize(vartype);
if( !tnode ){
   struct STable* nnode = (struct STable *)malloc(sizeof(struct STable));
   strcpy(nnode->label ,lab);
   strcpy(nnode->type ,vartype);
   nnode->size = size;
   nnode->location = 100;
   nnode->next=NULL;
   head = nnode;

}else{
   while(tnode->next){
      if(strcmp (tnode->label,lab) == 0){
         printf("\nError: ReDeclaration of %s Variable %s (Previous Declaration as %s)",vartype,
lab,tnode->type);
         return;
      }
      tnode = tnode->next;
   }
   struct STable* nnode = (struct STable *)malloc(sizeof(struct STable));
   strcpy(nnode->label ,lab);
   nnode->size = size;
   strcpy(nnode->type ,vartype);
   nnode->location = tnode -> location + size;
   nnode->next=NULL;
   tnode->next = nnode;
}
}
```

CD LAB 6 Semantic Analyzer using YACC

```
void DisplaySTable(struct STable *st){
int i = 1;
printf("\n\n\t\t\t\t%s\n","SYMBOL TABLE");
printf("\t| %s | Label | size | location |\n","Index");
while(st){
    printf("\t|%7d|%7s|%6d|%10d|\n",i++,st->label,st->size,st->location);
    st = st->next;
}
}
```

lex.l

```
%{
#include <stdio.h>
#include "y.tab.h"
%}

letter [a-zA-Z]
digit [0-9]

%%
"int" {return INT;}
"float" {return FLOAT;}
"double" {return DOUBLE;}
"char" {return CHAR;}
{letter}({letter}|{digit})* { yylval.SM.label = yytext; return ID;}
","|";"|"="|"+"|"-" {return yytext[0];}
\n
%%
```

CD LAB 6 Semantic Analyzer using YACC

INPUT

```
int bh,sa;
char as,g,sd;
double s1,h4;
float ff;

bh = sa + as;
bh = bh + bh;
s1 = ff;
asd;
asd = as + d2;
```

OUTPUT

```
Error: Type Mismatch INT and CHAR
Error: Type Mismatch DOUBLE and FLOAT
Error: Undeclared Variable asd

                         SYMBOL TABLE
        | Index | Label | size | location |
        |     1|    bh|    4|       100|
        |     2|    sa|    4|       104|
        |     3|    as|    1|       105|
        |     4|     g|    1|       106|
        |     5|    sd|    1|       107|
        |     6|    s1|    8|       115|
        |     7|    h4|    8|       123|
        |     8|    ff|    4|       127|
```