Name: Tushar Panchal

En.No: 21162101014

Sub: CD(Compiler Design)

Branch: CBA

Batch:71

------------------------------PRACTICAL 09------------------------------

YACC Program for C Variable Declaration

✓ *P1.l:*

```
%{
#include "p1.tab.h"
#include <string.h> /* Include this for strdup */
%}

%%
"int"            { return INT; }
"float"          { return FLOAT; }
"char"           { return CHAR; }
[ \t\n]+         ; /* Ignore whitespace */
[a-zA-Z_][a-zA-Z0-9_]* {
                    yylval.str = strdup(yytext);
                    return ID;
                 }
";"              { return SEMICOLON; }
","              { return COMMA; }
%%

int yywrap(void) {
    return 1;
}
```

✓ *P1.y:*

```
%{
#include <stdio.h>
#include <stdlib.h> /* For malloc, free */
```

```
#include <string.h> /* For strdup */
int yylex(void);
void yyerror(const char *str) {
    fprintf(stderr, "error: %s\n", str);
}
%}

%union {
    char *str; /* Member to hold strings */
}

%token <str> ID /* ID uses the str member */
%token INT FLOAT CHAR SEMICOLON COMMA

%%
declaration:
    type_specifier variable_list SEMICOLON {
        printf("Valid declaration.\n");
    }
;

type_specifier:
    INT {
        printf("Type: int\n");
    }
    | FLOAT {
        printf("Type: float\n");
    }
    | CHAR {
        printf("Type: char\n");
    }
;

variable_list:
    variable_list COMMA ID {
        printf("Variable: %s\n", $3);
        free($3);
    }
    | ID {
        printf("Variable: %s\n", $1);
        free($1);
    }
;
%%

int main(void) {
    return yyparse();
}
```

3

✓ *Output:*