CS358 - ASSIGNMENT 9

220002018 - Arnav Jain

Question 1

q1.l:

```
q1.l
%{
#include <stdio.h>
#include "q1.tab.h"
#include <string.h>
%}

%%
[a-zA-Z0-9]+ {
    strcpy(yylval.str, yytext);
    return STRING;
}
\n          { return 0; }
.           { /* Ignore all other characters */ }
%%

int yywrap() {
    return 1;
}

```

q1.y:

q1.y

```
%{
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int is_palindrome(char *str);
%}

%union {
    char str[100];
}

%token <str> STRING
%type <str> E

%%

S : E {
    if(is_palindrome($1))
        printf("\"%s\" is a palindrome.\n", $1);
    else
        printf("\"%s\" is not a palindrome.\n", $1);
}
;

E : STRING { strcpy($$, $1); }
;

%%

int is_palindrome(char *str) {
    int i, j;
    int len = strlen(str);

    // Convert to lowercase for case-insensitive comparison
    for(i = 0; i < len; i++)
        str[i] = tolower(str[i]);

    for(i = 0, j = len - 1; i < j; i++, j--) {
        if(str[i] != str[j])
            return 0;
    }
    return 1;
}
```

```
45
46  int main() {
47      printf("Enter a string to check if it's a palindrome: ");
48      yyparse();
49      return 0;
50  }
51
52  int yyerror(char *s) {
53      printf("Error: %s\n", s);
54      return 0;
55  }
```

Output:

```
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9$ ./q1
Enter a string to check if it's a palindrome: aaa
"aaa" is a palindrome.
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9$ ./q1
Enter a string to check if it's a palindrome: abb
"abb" is not a palindrome.
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9$ ./q1
Enter a string to check if it's a palindrome: aba
"aba" is a palindrome.
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9$
```

Question 2

q2.l:

```
ques 2 > ≡ q2.l
  1   %{
  2   #include "q2.tab.h"
  3   #include <stdlib.h>
  4   %}
  5
  6   %%
  7   [0-9]+       { yylval.num = atoi(yytext); return NUMBER; }
  8   "+"          { return PLUS; }
  9   "-"          { return MINUS; }
 10   "*"          { return TIMES; }
 11   "/"          { return DIVIDE; }
 12   "("          { return LPAREN; }
 13   ")"          { return RPAREN; }
 14   [ \t]        { /* Ignore whitespace */ }
 15   \n           { return 0; }
 16   .            { printf("Unexpected character: %s\n", yytext); }
 17   %%
 18
 19   int yywrap() {
 20       return 1;
 21   }
```

q2.y:

ques 2 > q2.y

```
%{
#include <stdio.h>
%}

%union {
    int num;
}

%token <num> NUMBER
%token PLUS MINUS TIMES DIVIDE LPAREN RPAREN
%left PLUS MINUS
%left TIMES DIVIDE
%right UMINUS

%type <num> E T F

%%

S : E { printf("Result: %d\n", $1); }
;

E : E PLUS T    { $$ = $1 + $3; }
  | E MINUS T   { $$ = $1 - $3; }
  | T           { $$ = $1; }
;

T : T TIMES F   { $$ = $1 * $3; }
  | T DIVIDE F  {
        if ($3 == 0) {
            yyerror("Division by zero");
            $$ = 0;
        } else {
            $$ = $1 / $3;
        }
    }
  | F           { $$ = $1; }
;

F : NUMBER      { $$ = $1; }
  | LPAREN E RPAREN { $$ = $2; }
  | MINUS F     { $$ = -$2; }
;

%%
```

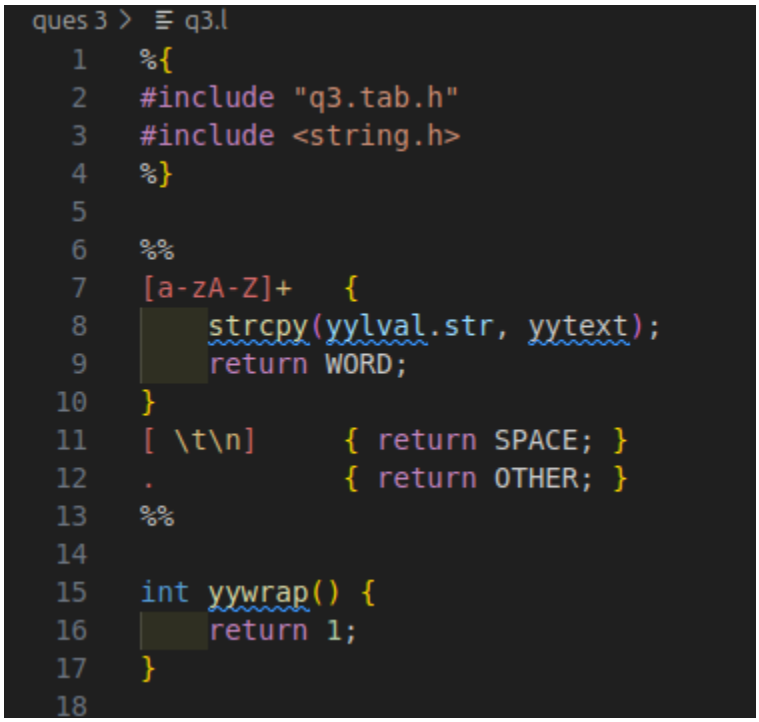
```
46  int main() {
47      printf("Enter an arithmetic expression: ");
48      yyparse();
49      return 0;
50  }
51
52  int yyerror(char *s) {
53      printf("Error: %s\n", s);
54      return 0;
55  }
```

Output:

```
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9/ques 2$ ./q2
Enter an arithmetic expression: 2+3*5
Result: 17
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9/ques 2$ ./q2
Enter an arithmetic expression: (2+3)*5
Result: 25
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9/ques 2$
```

Question 3

Q3.l:



Q3.y:

ques 3 > q3.y

```
%{
#include <stdio.h>
#include <string.h>
#include <ctype.h>

char target_word[100];
int count = 0;

void to_lowercase(char *str);
%}

%union {
    char str[100];
}

%token <str> WORD
%token SPACE OTHER
%type <str> S

%%

S : /* empty */
    | S WORD  {
        char temp[100];
        strcpy(temp, $2);
        to_lowercase(temp);

        if(strcmp(temp, target_word) == 0)
            count++;
    }
  | S SPACE { /* Ignore spaces */ }
  | S OTHER { /* Ignore other characters */ }
;

%%

void to_lowercase(char *str) {
    int i;
    for(i = 0; str[i]; i++)
        str[i] = tolower(str[i]);
}
```

```c
int main() {
    printf("Enter the word to count: ");
    scanf("%s", target_word);

    // Convert target word to lowercase
    to_lowercase(target_word);

    printf("Enter the text (Ctrl+D to end on Unix, Ctrl+Z on Windows):\n");

    // Clear input buffer
    while(getchar() != '\n');

    yyparse();

    printf("The word \"%s\" appears %d times in the text.\n", target_word, count);
    return 0;
}

int yyerror(char *s) {
    printf("Error: %s\n", s);
    return 0;
}

```

Output:

```
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9/ques 3$ ./q3
Enter the word to count: hello
Enter the text (Ctrl+D to end on Unix, Ctrl+Z on Windows):
hello hello helloThe word "hello" appears 3 times in the text.
arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiller-Techniques/LAB 9/ques 3$
```

Question 4

Q4.l

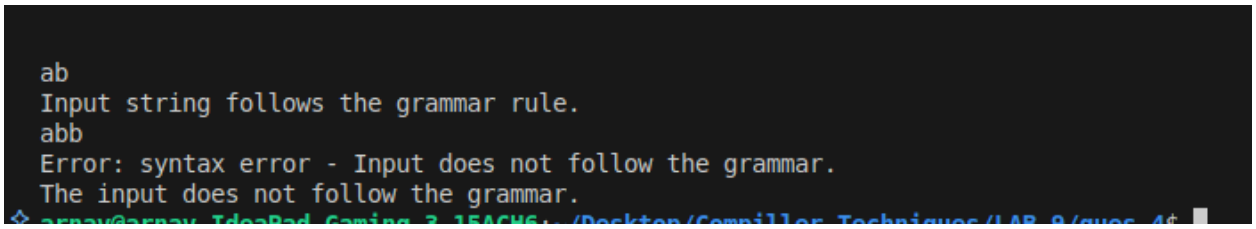
```
ques 4 > ≡ q4.l
  1   %{
  2   #include "q4.tab.h"
  3   %}
  4
  5   %%
  6   "a"          { return A; }
  7   "b"          { return B; }
  8   [ \t\n]      { /* Ignore whitespace */ }
  9   .            { return yytext[0]; }
 10   %%
 11
 12   int yywrap() {
 13       return 1;
 14   }
```

Q4.y

ques 4 > q4.y

```
%{
#include <stdio.h>
#include <stdlib.h>
%}

/* Tokens for the grammar */
%token A B

%%
/* Grammar: A -> aAb | bBa | ε */

start : expr  { printf("Input string follows the grammar rule.\n"); }
      ;

expr : /* epsilon */
     A expr B       /* This represents aAb */
     | B expr_b A     /* This represents bBa */
     |
     ;

expr_b : /* epsilon */
     |    /* This represents aAb */
     ;

%%

int main() {
    printf("Enter a string to check if it follows the grammar:\n");
    printf("A -> aAb | bBa | ε\n");
    printf("B -> ε\n");

    if(yyparse() == 0)
        printf("Parsing completed successfully.\n");
    else
        printf("The input does not follow the grammar.\n");

    return 0;
}

int yyerror(char *s) {
    printf("Error: %s - Input does not follow the grammar.\n", s);
    return 0;
}
```

Output



GitHub

https://github.com/arnavjain2710/Compiller-Techniques/tree/main/LAB%209