

CS358 - ASSIGNMENT 9

220002018 - Arnav Jain

Question 1

q1.l:

```
q1.l
1  %{
2  #include <stdio.h>
3  #include "q1.tab.h"
4  #include <string.h>
5  %}
6
7  %%
8  [a-zA-Z0-9]+ {
9      strcpy(yylval.str, yytext);
10     return STRING;
11 }
12 \n          { return 0; }
13 .          { /* Ignore all other characters */ }
14 %%
15
16 int yywrap() {
17     return 1;
18 }
19
```

q1.y:

≡ q1.y

```
1  %{
2  #include <stdio.h>
3  #include <string.h>
4  #include <ctype.h>
5
6  int is_palindrome(char *str);
7  %}
8
9  %union {
10     char str[100];
11 }
12
13 %token <str> STRING
14 %type <str> E
15
16 %%
17
18 S : E {
19     if(is_palindrome($1))
20         printf("\"%s\" is a palindrome.\n", $1);
21     else
22         printf("\"%s\" is not a palindrome.\n", $1);
23 }
24 ;
25
26 E : STRING { strcpy($$, $1); }
27 ;
28
29 %%
30
31 int is_palindrome(char *str) {
32     int i, j;
33     int len = strlen(str);
34
35     // Convert to lowercase for case-insensitive comparison
36     for(i = 0; i < len; i++)
37         str[i] = tolower(str[i]);
38
39     for(i = 0, j = len - 1; i < j; i++, j--) {
40         if(str[i] != str[j])
41             return 0;
42     }
43     return 1;
44 }
```

```

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/Compiler-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/Compiler-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/Compiler-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/Compiler-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]+      { yyval.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

```
1  %{
2  #include <stdio.h>
3  %}
4
5  %union {
6  int num;
7  }
8
9  %token <num> NUMBER
10 %token PLUS MINUS TIMES DIVIDE LPAREN RPAREN
11 %left PLUS MINUS
12 %left TIMES DIVIDE
13 %right UMINUS
14
15 %type <num> E T F
16
17 %%
18
19 S : E { printf("Result: %d\n", $1); }
20 ;
21
22 E : E PLUS T    { $$ = $1 + $3; }
23   | E MINUS T   { $$ = $1 - $3; }
24   | T           { $$ = $1; }
25 ;
26
27 T : T TIMES F    { $$ = $1 * $3; }
28   | T DIVIDE F   {
29       if ($3 == 0) {
30         yyerror("Division by zero");
31         $$ = 0;
32       } else {
33         $$ = $1 / $3;
34       }
35   }
36   | F           { $$ = $1; }
37 ;
38
39 F : NUMBER       { $$ = $1; }
40   | LPAREN E RPAREN { $$ = $2; }
41   | MINUS F      { $$ = -$2; }
42 ;
43
44 %%
```

```

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

```

Question 3

Q3.l:

```
ques 3 > ≡ q3.l
1  %{
2  #include "q3.tab.h"
3  #include <string.h>
4  %{
5
6  %%
7  [a-zA-Z]+  {
8      strcpy(yyval.str, yytext);
9      return WORD;
10 }
11 [ \t\n]    { return SPACE; }
12 .          { return OTHER; }
13 %%
14
15 int yywrap() {
16     return 1;
17 }
18
```

Q3.y:

ques 3 > ≡ q3.y

```
1  %{
2  #include <stdio.h>
3  #include <string.h>
4  #include <ctype.h>
5
6  char target_word[100];
7  int count = 0;
8
9  void to_lowercase(char *str);
10 %}
11
12 %union {
13     char str[100];
14 }
15
16 %token <str> WORD
17 %token SPACE OTHER
18 %type <str> S
19
20 %%
21
22 S : /* empty */
23     | S WORD {
24         char temp[100];
25         strcpy(temp, $2);
26         to_lowercase(temp);
27
28         if(strcmp(temp, target_word) == 0)
29             count++;
30     }
31     | S SPACE { /* Ignore spaces */ }
32     | S OTHER { /* Ignore other characters */ }
33 ;
34
35 %%
36
37 void to_lowercase(char *str) {
38     int i;
39     for(i = 0; str[i]; i++)
40         str[i] = tolower(str[i]);
41 }
```



```

43  int main() {
44      printf("Enter the word to count: ");
45      scanf("%s", target_word);
46
47      // Convert target word to lowercase
48      to_lowercase(target_word);
49
50      printf("Enter the text (Ctrl+D to end on Unix, Ctrl+Z on Windows):\n");
51
52      // Clear input buffer
53      while(getchar() != '\n');
54
55      yyparse();
56
57      printf("The word \"%s\" appears %d times in the text.\n", target_word, count);
58      return 0;
59  }
60
61  int yyerror(char *s) {
62      printf("Error: %s\n", s);
63      return 0;
64  }
65

```

Output:

```

• arnav@arnav-IdeaPad-Gaming-3-15ACH6:~/Desktop/Compiler-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/Compiler-Techniques/LAB 9/ques 3$ █

```

Question 4

Q4.1

```
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
1  %{
2  #include <stdio.h>
3  #include <stdlib.h>
4  %}
5
6  /* Tokens for the grammar */
7  %token A B
8
9  %%
10 /* Grammar: A -> aAb | bBa | ε */
11
12 start : expr { printf("Input string follows the grammar rule.\n"); }
13 ;
14
15 expr : /* epsilon */
16      A expr B      /* This represents aAb */
17      | B expr_b A   /* This represents bBa */
18      |
19      ;
20
21 expr_b : /* epsilon */
22         | /* This represents aAb */
23         ;
24
25 %%
26
27 int main() {
28     printf("Enter a string to check if it follows the grammar:\n");
29     printf("A -> aAb | bBa | ε\n");
30     printf("B -> ε\n");
31
32     if(yyparse() == 0)
33         printf("Parsing completed successfully.\n");
34     else
35         printf("The input does not follow the grammar.\n");
36
37     return 0;
38 }
39
40 int yyerror(char *s) {
41     printf("Error: %s - Input does not follow the grammar.\n", s);
42     return 0;
43 }

```

Output

```
ab
Input string follows the grammar rule.
abb
Error: syntax error - Input does not follow the grammar.
The input does not follow the grammar.
✚ arnav@arnav-IdeaPad-Gaming-3-15ACN6: /Desktop/Compiler-Techniques/LAB 8/ques 4$
```

GitHub

<https://github.com/arnavjain2710/Compiler-Techniques/tree/main/LAB%209>