

## MINI ASSIGNMENT - 1 - LEX

CS22BTECH11012 - Bolla Nehasree

CS24RESCH11010 - C. Sree Vyshnavi

1.

### Issues with the Given Lex Program

The provided lex program has a few issues:

- **Handling of Multi-line Comments:** The rule for multi-line comments `{start}(\n|.)*{end}` is too greedy. It will match everything from the first occurrence of `/*` to the last occurrence of `*/` in the entire input. This means it might not correctly handle nested or multiple comment blocks.
- **Printing everything except comments:** No rule in the lex program prints everything except the comments.

### Example

```
/* Start of the comment
  Still inside the comment */

int main() {
    // This is a single-line comment
    printf("Hello, World!"); /* Another comment */
}
```

The given lex program might match the text from the first `/*` to the last `*/` in the entire file and thus removing too much content and not handling the comments correctly.

## 2.

### Corrected LEX Program

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

start \\\*
end \*\V

%%
\\V.*      { /* This pattern matches C-style comments */ }
{start}([^\*]\\*+[/])*\{end} { /* This pattern matches C++-style comments */ }
.\|\\n     { printf("%s", yytext); }
%%

int yywrap() {
    return 1;
}

int main(int argc, char **argv) {
    if (argc != 2) {
        fprintf(stderr, "Usage: %s <filename>\n", argv[0]);
        return 1;
    }

    FILE *file = fopen(argv[1], "r");
    if (!file) {
        perror("fopen");
        return 1;
    }

    yyin = file;
    yylex();
    fclose(file);
    return 0;
}
```

### 3.

#### Code:

```
%{
    int count = 0;
    int sum = 0;
    int x = 0;
}%

%%
[0-9]+ {
    x = atoi(yytext);
    printf("integer: %d\n", x);
    count++;
    sum += x;
}
.      ;
\n     ;
%%

int yywrap(){

int main(int argc, char *argv[]) {
    //cmd line arg to take input from file
    if (argc != 2) {
        fprintf(stderr, "Usage: %s <filename>\n", argv[0]);
        return 1;
    }
    FILE *file = fopen(argv[1], "r");
    if (!file) {
        perror("fopen");
        return 1;
    }
    yyin = file;

    yylex();
    printf("count = %d, sum = %d\n", count, sum);
    float avg = (float)sum/(float)count;
    printf("\nAverage of the integers = %f\n", avg);

    return 0;
}
```

#### 4. Warning: Rule cannot be matched

The warning “rule cannot be matched” occurs when a rule is already covered by another rule specified before it.

For example, `[A-Z]+` (line 11) is already covered in the previous rule `[A-Za-z]+` (line 10). Hence, this program throws a warning in line 11

#### CODE EXAMPLE:

```
%%  
[A-Za-z]+ {printf("string1: %s\n", yytext);}  
[A-Z]+ {printf("string2: %s\n", yytext); /* THROWS WARNING */}  
.  
\n  
%%  
  
int yywrap(){  
  
int main(int argc, char *argv[]) {  
    yylex();  
    return 0;  
}
```