Program A: Check if an Email ID is Valid

**email\_check.l**

%{

#include <stdio.h>

%}

%%

[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.(com|org|net|edu|gov|in|co|io) {

printf("Valid Email ID: %s\n", yytext);

}

[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,} {

printf("Invalid Email: %s (unsupported domain)\n", yytext);

}

.|\n {

printf("Invalid Input: %s\n", yytext);

}

%%

int main() {

printf("Enter an email address:\n");

yylex();

return 0;

}

ProgramB: Check if a Statement is Simple or Compound

statement\_check.l

%{

#include <stdio.h>

#include <string.h>

int compound = 0;

%}

%%

// Conjunctions or markers of compound/complex sentences

(and|but|because|although|however|if|unless|until|while|when|since|though|or|yet) {

compound = 1;

}

// Words, punctuation

[a-zA-Z]+ { /\* word \*/ }

[.,;!?] { /\* punctuation \*/ }

// Newline or end of statement

\n {

if (compound)

printf("This is a COMPOUND statement.\n");

else

printf("This is a SIMPLE statement.\n");

compound = 0; // Reset for next input

}

// Any other characters

. { /\* ignore \*/ }

%%

int main() {

printf("Enter a sentence (press Enter to submit each one):\n");

yylex();

return 0;

}

Program C: Eliminate Comments

remove\_comments.l

%{

#include <stdio.h>

%}

%%

// -------- Remove C++-style single-line comments --------

"//".\* { /\* skip comment \*/ }

// -------- Remove C-style multi-line comments --------

"/\*"([^\*]|\\*+[^\*/])\*\\*+"/" { /\* skip multi-line comment \*/ }

// -------- Print everything else --------

.|\n { printf("%s", yytext); }

%%

int main() {

printf("Enter code (with comments). Press Ctrl+Z (Windows) or Ctrl+D (Linux/Mac) to finish:\n");

yylex();

return 0;

}

Output:

Sample Input:

int main() {

// this is a single-line comment

printf("Hello"); /\* this is an

inline comment \*/

return 0;

}

Output (comments removed):

int main() {

printf("Hello");

return 0;

}