Develop a lexical Analyzer to identify identifiers, constants, operators using C program?

#include <stdio.h>

#include <ctype.h>

#include <string.h>

void check\_token(char \*str) {

if (isalpha(str[0]) || str[0] == '\_') {

// It's an identifier

int i = 0;

while (str[i] != '\0') {

if (!isalnum(str[i]) && str[i] != '\_') {

printf("Invalid identifier: %s\n", str);

return;

}

i++;

}

printf("Identifier: %s\n", str);

}

else if (isdigit(str[0])) {

int i = 0;

while (str[i] != '\0') {

if (!isdigit(str[i])) {

printf("Invalid constant: %s\n", str);

return;

}

i++;

}

printf("Constant: %s\n", str);

}

else {

if (str[0] == '+' || str[0] == '-' || str[0] == '\*' || str[0] == '/' ||

str[0] == '=' || str[0] == '<' || str[0] == '>') {

printf("Operator: %s\n", str);

} else {

printf("Unknown: %s\n", str);

}

}

}

int main() {

char input[100];

printf("Enter a string: ");

fgets(input, sizeof(input), stdin);

char token[20];

int i = 0, j = 0;

while (input[i] != '\0') {

if (isspace(input[i])) {

// Skip whitespace characters

i++;

continue;

}

if (isalnum(input[i]) || input[i] == '\_') {

token[j++] = input[i++];

token[j] = '\0'; // Null-terminate the token

check\_token(token);

j = 0;

}

else if (strchr("+-\*/=<>", input[i])) {

token[j++] = input[i++];

token[j] = '\0'; // Null-terminate the token

check\_token(token);

j = 0;

} else {

i++;

}

}

return 0;

}

