#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

#define MAX\_IDENTIFIER\_LENGTH 32

int is\_identifier\_char(char c) {

return isalnum(c) || c == '\_';

}

int is\_valid\_identifier(const char \*str) {

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

return 0;

}

for (int i = 1; str[i] != '\0'; i++) {

if (!is\_identifier\_char(str[i])) {

return 0;   
 }

}

return 1;

}

int main() {

char input[100];

printf("Enter the input string: ");

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

input[strcspn(input, "\n")] = 0;

char \*token;

char \*rest = input;

while ((token = strtok\_r(rest, " \t+-\*/=();", &rest)) != NULL) {   
 if (strlen(token) == 0) continue;

if (isdigit(token[0])) {

int is\_constant = 1;

for(int i = 0; i < strlen(token); i++){

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

is\_constant = 0;

break;

}

}

if(is\_constant){

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

} else {

printf("Invalid Token: %s\n", token);

}

} else if (isalpha(token[0]) || token[0] == '\_') {

if (is\_valid\_identifier(token)) {

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

} else {

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

}

} else if (strchr("+-\*/=();", token[0]) != NULL) {

printf("Operator/Punctuation: %s\n", token);

} else {

printf("Invalid Token: %s\n", token);

}

}

return 0;

}