3.Design a lexical Analyzer for given language should ignore the redundant spaces, tabs and new lines and ignore comments using C

#include <stdio.h>

#include <ctype.h>

#include <string.h>

void removeCommentsAndWhitespace(char \*source);

int main() {

char code[] = "int main() { /\* This is a comment \*/\n"

" int a = 5; // Another comment\n"

" float b = 2.5;\n"

"}\n";

printf("Original Code:\n%s\n", code);

removeCommentsAndWhitespace(code);

printf("Processed Code:\n%s\n", code);

return 0;

}

void removeCommentsAndWhitespace(char \*source) {

char \*i = source, \*j = source;

while (\*i) {

if (\*i == '/' && \*(i + 1) == '/') {

while (\*i && \*i != '\n') i++;

} else if (\*i == '/' && \*(i + 1) == '\*') {

while (\*i && !(\*i == '\*' && \*(i + 1) == '/')) i++;

i += 2;

} else if (\*i == ' ' || \*i == '\t' || \*i == '\n') {

i++;

} else {

\*j++ = \*i++;

}

}

\*j = '\0';

}

