```c
/*
********************************************************************************
*                            countSentences
********************************************************************************
Command format:

            countSentences  <input.txt>  <output.txt>

Counts the number of sentences between 2 full stops.

When a '.' is found, the next character is checked and if this is an empty space
(blank, new line, or tab) and that's considered the end of a sentence. Same if
EOF follows the '.'.
In the case of a file with no text, a message is printed in the output file.

N.B. i is initialized to 1 to properly count sentences.
N.B. Everything is done inside main because the task is very simple.
*/


# include <stdio.h>
# include <ctype.h>

# define MAX 1000



int main (int argc, char * argv[]) {
    if (argc != 3) {
        printf ("\n Incorrect Command!\n");
        return 0;
    }

    FILE * fin, * fout;
    int c;

    fin  = fopen (*(argv + 1), "r");
    fout = fopen (*(argv + 2), "w");

    /* Skip every initial non-alphanumeric character */
    while ( !isalnum( (c = getc(fin)) ))
        ;

    if (c == EOF) {
        fprintf (fout , "<%s> is empty.\n" , *(argv + 1));
        printf("\nResult in %s" , *(argv + 2));
        return 0;
    }

    int i = 0;

    while ((c = getc(fin)) != EOF) {
        if (c == '.')
            if ( isspace( (c = getc(fin)) ) || c == EOF )
                i++;
    }



    fprintf (fout, "Number of sentences: %d\n" , i);
    printf("\nResults in %s\n" , *(argv + 2));

    fclose (fin);
    fclose (fout);
}
```