```c
#include <stdio.h>
#include <string.h>

#define MAX_FILENAME_LENGTH 100
#define MAX_LINE_LENGTH 1000

int main() {
    char filename[MAX_FILENAME_LENGTH];
    char search_string[MAX_LINE_LENGTH];

    printf("Enter the name of the file: ");
    scanf("%s", filename);
    printf("Enter the search string: ");
    scanf(" %[^\n]", search_string);

    FILE *file = fopen(filename, "r");
    char line[MAX_LINE_LENGTH];

    if (file == NULL) {
        printf("Error opening file\n");
        return 1;
    }

    while (fgets(line, sizeof(line), file)) {
        if (strstr(line, search_string) != NULL) {
            printf("%s", line);
        }
    }
}
```

Enter the name of the file: index
Enter the search string: 1
Error opening file

Process returned 1 (0x1)   execution time : 22.967 s
Press any key to continue.

Checking for existence: C:\Users\kmoha\OneDrive\Documents\assi2\bin\Debug\assi2.exe
Set variable: PATH=.;C:\Program Files\CodeBlocks\MinGW\bin;C:\Program Files\CodeBlocks\Mir
\OpenSSH;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\Program Files\NVIDIA Co
\AppData\Local\Programs\Python\Python310;C:\Users\kmoha\AppData\Local\Microsoft\WindowsApp
Executing: "C:\Program Files\CodeBlocks\cb_console_runner.exe" "C:\Users\kmoha\OneDrive\Dc

```c
typedef struct {
    int id;
    char name[20];
    float price;
} Product;

int main() {
    FILE *file = fopen("data.bin", "rb");
    Product product;

    if (file == NULL) {
        printf("Error opening file\n");
        return 1;
    }

    while (fread(&product, sizeof(product), 1, file)) {
        printf("ID: %d\n", product.id);
        printf("Name: %s\n", product.name);
        printf("Price: %.2f\n\n", product.price);
    }

    fclose(file);

    return 0;
}
```

Error opening file

Process returned 1 (0x1)   execution time : 0.024 s
Press any key to continue.

```c
#include <stdio.h>
#include <string.h>

#define MAX_LINE_LENGTH 1000

int main() {
    FILE *input_file = fopen("input.txt", "r");
    FILE *output_file = fopen("output.txt", "w");
    char line[MAX_LINE_LENGTH];

    if (input_file == NULL || output_file == NULL) {
        printf("Error opening file\n");
        return 1;
    }

    while (fgets(line, sizeof(line), input_file)) {
        char *word = strtok(line, " ");
        while (word != NULL) {
            if (strcmp(word, "red\n") == 0) {
                fprintf(output_file, "blue ");
            } else {
                fprintf(output_file, "%s ", word);
            }
            word = strtok(NULL, " ");
        }
    }

    fclose(input_file);
```

Logs & others

Code::Blocks ✕   Search results ✕   Cccc ✕   Build log ✕   Build messages ✕   CppCheck/Vera++ ✕   CppCheck/Vera++ messages ✕

```
Checking for existence: C:\Users\kmoha\OneDrive\Documents\assi2\bin\Debug\assi2.exe
Set variable: PATH=.;C:\Program Files\CodeBlocks\MinGW\bin;C:\Program Files\CodeBlocks\MinGW;C:\Windows\System32;C:\Windows;C:\Windows\Syste
\OpenSSH;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Program Files\dotnet;C
\AppData\Local\Programs\Python\Python310;C:\Users\kmoha\AppData\Local\Microsoft\WindowsApps;\;C:\Users\kmoha\AppData\Local\Programs\Microsof
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\kmoha\OneDrive\Documents\assi2\bin\Debug\assi2.exe"  (in C:\Users\k
```

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

#define NUM_LETTERS 26

int main() {
    FILE *input_file = fopen("data.txt", "r");
    FILE *output_file = fopen("statistics.txt", "w");

    if (input_file == NULL || output_file == NULL) {
        printf("Error opening file\n");
        return 1;
    }

    int letter_counts[NUM_LETTERS] = {0};
    int ch;

    while ((ch = fgetc(input_file)) != EOF) {
        if (isalpha(ch)) {
            letter_counts[tolower(ch) - 'a']++;
        }
    }

    for (int i = 0; i < NUM_LETTERS; i++) {
        fprintf(output_file, "%c: %d\n", 'a' + i, letter_counts[i]);
    }

    fclose(input_file);
```

```c
#include <stdio.h>
#include <string.h>

int main() {
    FILE *input_file = fopen("input.txt", "r");
    FILE *error_log_file = fopen("error_log.txt", "w");
    char line[100];

    if (input_file == NULL || error_log_file == NULL) {
        printf("Error opening file\n");
        return 1;
    }

    while (fgets(line, sizeof(line), input_file)) {
        if (strstr(line, "error") != NULL) {
            fputs(line, error_log_file);
        }
    }

    fclose(input_file);
    fclose(error_log_file);

    error_log_file = fopen("error_log.txt", "r");

    if (error_log_file == NULL) {
        printf("Error opening file\n");
        return 1;
    }
}
```

Error opening file

Process returned 1 (0x1)   execution time : 0.010 s
Press any key to continue.