1- Write a program that counts the number of characters in a file, excluding spaces and newline characters. Create a file, write a sentence inside it, and then count the characters.

اكتب برنامجًا يحسب عدد الأحرف في الملف، باستثناء المسافات وأحرف السطر الجديد. قم بإنشاء ملف، واكتب جملة بداخله، ثم قم بعد الأحرف.

Input

sample.txt

```
1  This is a sample sentence.
```

Output

```
Count: 22
```

Solution

```c
// www.gammal.tech

#include<stdio.h>

int main() {
    FILE* file = fopen("sample.txt", "w");
    fprintf(file, "This is a sample sentence.");
    fclose(file);

    file = fopen("sample.txt", "r");
    int count = 0;
    char ch;

    while (fscanf(file, "%c", &ch) != EOF) {
        if (ch != ' ' && ch != '\n') {
            count++;
        }
    }

    printf("Count: %d\n", count);

    fclose(file);

    return 0;
}
```

2- Create a program that copies the content of one file to another file.

إنشاء برنامج يقوم بنسخ محتوى ملف إلى ملف آخر.

Input & Output

destination.txt

```
1  This is the source file content.
```

source.txt

```
1  This is the source file content.
```

Solution

```c
// www.gammal.tech

#include<stdio.h>

int main() {
    FILE* sourceFile = fopen("source.txt", "w");
    fprintf(sourceFile, "This is the source file content.");
    fclose(sourceFile);

    FILE* destinationFile = fopen("destination.txt", "w");

    sourceFile = fopen("source.txt", "r");
    char ch;

    while (fscanf(sourceFile, "%c", &ch) != EOF) {
        fprintf(destinationFile, "%c", ch);
    }

    fclose(sourceFile);
    fclose(destinationFile);

    return 0;
}
```

3- Write a program that reads a sentence from a file and prints each word on a new line.

اكتب برنامجًا يقرأ جملة من ملف ويطبع كل كلمة في سطر جديد.

Input

sentence.txt

```
1  This is a simple sentence.
```

Output

```
This
is
a
simple
sentence.
```

Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    FILE* file = fopen("sentence.txt", "w");
    fprintf(file, "This is a simple sentence.");
    fclose(file);

    file = fopen("sentence.txt", "r");
    char word[20];

    while (fscanf(file, "%s", word) != EOF) {
        printf("%s\n", word);
    }

    fclose(file);

    return 0;
}
```

4- Create a program that appends a new sentence to an existing file.

إنشاء برنامج يقوم بإلحاق جملة جديدة بملف موجود.

Output

existing_file.txt

```
1
2  This is a new sentence.
```

Solution

```c
// www.gammal.tech

#include<stdio.h>

int main() {
    FILE* file = fopen("existing_file.txt", "a");
    fprintf(file, "\nThis is a new sentence.");
    fclose(file);

    return 0;
}
```

5- Write a program that checks if a specific word exists in a file. If the word is found, print a message; otherwise, print another message.

اكتب برنامجًا يتحقق من وجود كلمة معينة في الملف. إذا تم العثور على الكلمة، طباعة رسالة؛ وإلا، قم بطباعة رسالة أخرى.

Input

```
Enter the word to search: word
Word found in the file.
```

Input & Output

```
Enter the word to search: word
Word found in the file.
```

Solution

```c
// www.gammal.tech

#include<stdio.h>

int main() {
    FILE* file = fopen("sample.txt", "r");
    char word[20], searchWord[20];

    printf("Enter the word to search: ");
    scanf("%s", searchWord);

    int found = 0;

    while (fscanf(file, "%s", word) != EOF) {
        if (strcmp(word, searchWord) == 0) {
            found = 1;
            break;
        }
    }

    if (found) {
        printf("Word found in the file.\n");
    } else {
        printf("Word not found in the file.\n");
    }

    fclose(file);

    return 0;
}
```

6- Create a program that reads a series of numbers from a file and prints their sum.

قم بإنشاء برنامج يقرأ سلسلة من الأرقام من ملف ويطبع مجموعها.

Input

numbers.txt

```
1   10 20 30 40 50
```

Output

```
Sum of numbers: 150
```

Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    FILE* numbersFile = fopen("numbers.txt", "w");
    fprintf(numbersFile, "10 20 30 40 50");
    fclose(numbersFile);

    numbersFile = fopen("numbers.txt", "r");
    int number, sum = 0;

    while (fscanf(numbersFile, "%d", &number) != EOF) {
        sum += number;
    }

    printf("Sum of numbers: %d\n", sum);

    fclose(numbersFile);

    return 0;
}
```

7- Write a program that appends a new number to an existing file containing a series of numbers.

اكتب برنامجًا يقوم بإلحاق رقم جديد بملف موجود يحتوي على سلسلة من الأرقام.

Output

existing_numbers.txt

```
1    60
```

Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    FILE* numbersFile = fopen("existing_numbers.txt", "a");
    fprintf(numbersFile, " 60");
    fclose(numbersFile);

    return 0;
}
```

8- Create a program that reads a file with a series of grades and calculates the average.

أنشئ برنامجًا يقرأ ملفًا يحتوي على سلسلة من الدرجات ويحسب المتوسط.

Input

grades.txt

```
1   85 92 78 95 88
```

Output

```
Average grade: 87.60
```

Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    FILE* gradesFile = fopen("grades.txt", "w");
    fprintf(gradesFile, "85 92 78 95 88");
    fclose(gradesFile);

    gradesFile = fopen("grades.txt", "r");
    int grade, sum = 0, count = 0;

    while (fscanf(gradesFile, "%d", &grade) != EOF) {
        sum += grade;
        count++;
    }

    if (count > 0) {
        double average = (double)sum / count;
        printf("Average grade: %.2lf\n", average);
    } else {
        printf("No grades found.\n");
    }

    fclose(gradesFile);

    return 0;
}
```

9- Write a program that reads a file containing prices of items and calculates the total cost.

اكتب برنامجاً يقرأ ملفاً يحتوي على أسعار الأصناف ويحسب التكلفة الإجمالية.

Input

prices.txt

```
1   15.50 20.75 8.99 12.45 30.25
```

Output

```
Total cost: 87.94
```

Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    FILE* pricesFile = fopen("prices.txt", "w");
    fprintf(pricesFile, "15.50 20.75 8.99 12.45 30.25");
    fclose(pricesFile);

    pricesFile = fopen("prices.txt", "r");
    double price, totalCost = 0;

    while (fscanf(pricesFile, "%lf", &price) != EOF) {
        totalCost += price;
    }

    printf("Total cost: %.2lf\n", totalCost);

    fclose(pricesFile);

    return 0;
}
```

10- Write a program that reads a file with a list of names and prints them in reverse order.

اكتب برنامجًا يقرأ ملفًا يحتوي على قائمة الأسماء ويطبعها بترتيب عكسي.

Input

names.txt

```
1  Alice Bob Charlie David Eve
```

Output

```
Names in reverse order:
Eve
David
Charlie
Bob
Alice
```

Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    FILE* namesFile = fopen("names.txt", "w");
    fprintf(namesFile, "Alice Bob Charlie David Eve");
    fclose(namesFile);

    namesFile = fopen("names.txt", "r");
    char name[20][20];
    int count = 0;

    while (fscanf(namesFile, "%s", name[count]) != EOF) {
        count++;
    }

    printf("Names in reverse order:\n");
    for (int i = count - 1; i >= 0; i--) {
        printf("%s\n", name[i]);
    }

    fclose(namesFile);

    return 0;
}
```
