**Day 9**

1. Write a program to create and write to a text file.

#include <stdio.h>

int main() {

FILE \*f = fopen("file.txt", "w");

if (f == NULL) {

printf("Error opening file.\n");

return 1;

}

fprintf(f, "Hello, file handling in C!");

fclose(f);

return 0;

}

2. Write a program to read contents of a file and display it.

#include <stdio.h>

int main() {

FILE \*f = fopen("file.txt", "r");

if (f == NULL) {

printf("File not found.\n");

return 1;

}

char ch;

while ((ch = fgetc(f)) != EOF) {

putchar(ch);

}

fclose(f);

return 0;

}

3. Write a program to count the number of lines in a file.

#include <stdio.h>

int main() {

FILE \*f = fopen("file.txt", "r");

int count = 0;

char ch;

while ((ch = fgetc(f)) != EOF) {

if (ch == '\n') count++;

}

fclose(f);

printf("Number of lines: %d\n", count + 1); // Add 1 for last line

return 0;

}

4. Write a program to copy contents from one file to another.

#include <stdio.h>

int main() {

FILE \*src = fopen("file.txt", "r");

FILE \*dest = fopen("copy.txt", "w");

char ch;

while ((ch = fgetc(src)) != EOF) {

fputc(ch, dest);

}

fclose(src);

fclose(dest);

printf("File copied successfully.\n");

return 0;

}

5. Write a program to append text to a file.

#include <stdio.h>

int main() {

FILE \*f = fopen("file.txt", "a");

fprintf(f, "\nThis is an appended line.");

fclose(f);

return 0;

}

6. Write a program to count the number of vowels in a file.

#include <stdio.h>

int main() {

FILE \*f = fopen("file.txt", "r");

char ch;

int count = 0;

while ((ch = fgetc(f)) != EOF) {

if (ch == 'a'|| ch == 'e'|| ch == 'i'|| ch == 'o'|| ch == 'u' ||

ch == 'A'|| ch == 'E'|| ch == 'I'|| ch == 'O'|| ch == 'U') {

count++;

}

}

fclose(f);

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

return 0;

}

7. Write a program to read integers from a file and find the sum.

#include <stdio.h>

int main() {

FILE \*f = fopen("numbers.txt", "r");

int num, sum = 0;

while (fscanf(f, "%d", &num) != EOF) {

sum += num;

}

fclose(f);

printf("Sum = %d\n", sum);

return 0;

}

8. Write a program to read a structure from a file.

#include <stdio.h>

struct Student {

char name[50];

int age;

};

int main() {

FILE \*f = fopen("student.dat", "rb");

struct Student s;

fread(&s, sizeof(s), 1, f);

printf("Name: %s\nAge: %d\n", s.name, s.age);

fclose(f);

return 0;

}

9. Write a program to sort names stored in a file.

#include <stdio.h>

#include <string.h>

int main() {

FILE \*f = fopen("names.txt", "r");

char names[100][50];

int count = 0;

while (fgets(names[count], 50, f)) {

names[count][strcspn(names[count], "\n")] = 0; // Remove newline

count++;

}

fclose(f);

// Bubble sort

char temp[50];

for (int i = 0; i < count-1; i++) {

for (int j = i+1; j < count; j++) {

if (strcmp(names[i], names[j]) > 0) {

strcpy(temp, names[i]);

strcpy(names[i], names[j]);

strcpy(names[j], temp);

}

}

}

printf("Sorted names:\n");

for (int i = 0; i < count; i++) {

printf("%s\n", names[i]);

}

return 0;

}

10. Write a program to search for a word in a file.

#include <stdio.h>

#include <string.h>

int main() {

FILE \*f = fopen("file.txt", "r");

char word[50], search[50] = "file";

int found = 0;

while (fscanf(f, "%s", word) != EOF) {

if (strcmp(word, search) == 0) {

found = 1;

break;

}

}

fclose(f);

if (found)

printf("Word found.\n");

else

printf("Word not found.\n");

return 0;

}