Assignment (11/08/2025)

Reg No : 192521112

Name : Vohas Vijayakumar Jayanthi

## **1. Create and Write to a Text File**

**IPO**

* **Input**: Text entered by user
* **Process**: Open file in write mode, write text to file
* **Output**: File created with text

#include <stdio.h>

int main() {

FILE \*fp;

char text[100];

fp = fopen("file1.txt", "w");

printf("Enter text: ");

fgets(text, sizeof(text), stdin);

fputs(text, fp);

fclose(fp);

printf("File created and written successfully.\n");

return 0;

}

**Sample Output**

Enter text: Hello File Handling

File created and written successfully.

## **2. Read Contents of a File and Display**

**IPO**

* **Input**: File name
* **Process**: Open file in read mode, display content
* **Output**: Content of file

#include <stdio.h>

int main() {

FILE \*fp;

char ch;

fp = fopen("file1.txt", "r");

if (fp == NULL) {

printf("File not found!");

return 1;

}

while ((ch = fgetc(fp)) != EOF)

putchar(ch);

fclose(fp);

return 0;

}

**Sample Output**

Hello File Handling

## **3. Count Number of Lines in a File**

**IPO**

* **Input**: File name
* **Process**: Count '\n' characters
* **Output**: Number of lines

#include <stdio.h>

int main() {

FILE \*fp;

char ch;

int count = 0;

fp = fopen("file1.txt", "r");

if (fp == NULL) {

printf("File not found!");

return 1;

}

while ((ch = fgetc(fp)) != EOF)

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

count++; // last line

fclose(fp);

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

return 0;

}

**Sample Output**

Total lines: 1

## **4. Copy Contents from One File to Another**

**IPO**

* **Input**: Source file and destination file
* **Process**: Read from source, write to destination
* **Output**: Destination file with same content

#include <stdio.h>

int main() {

FILE \*fs, \*fd;

char ch;

fs = fopen("file1.txt", "r");

fd = fopen("file2.txt", "w");

while ((ch = fgetc(fs)) != EOF)

fputc(ch, fd);

fclose(fs);

fclose(fd);

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

return 0;

}

**Sample Output**

File copied successfully.

## **5. Append Text to a File**

**IPO**

* **Input**: Text from user
* **Process**: Open file in append mode, write text
* **Output**: Text added to existing file

#include <stdio.h>

int main() {

FILE \*fp;

char text[100];

fp = fopen("file1.txt", "a");

printf("Enter text to append: ");

fgets(text, sizeof(text), stdin);

fputs(text, fp);

fclose(fp);

printf("Text appended successfully.\n");

return 0;

}

**Sample Output**

Enter text to append: More data here

Text appended successfully.

## **6. Count Vowels in a File**

**IPO**

* **Input**: File name
* **Process**: Check each char if vowel
* **Output**: Count of vowels

#include <stdio.h>

#include <ctype.h>

int main() {

FILE \*fp;

char ch;

int count = 0;

fp = fopen("file1.txt", "r");

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

ch = tolower(ch);

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

count++;

}

fclose(fp);

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

return 0;

}

**Sample Output**

Vowels: 7

## **7. Read Integers from File and Find Sum**

**IPO**

* **Input**: Integers in file
* **Process**: Read and sum integers
* **Output**: Sum

#include <stdio.h>

int main() {

FILE \*fp;

int num, sum = 0;

fp = fopen("numbers.txt", "r");

while (fscanf(fp, "%d", &num) == 1)

sum += num;

fclose(fp);

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

return 0;

}

**Sample Output**

Sum = 45

## **8. Read a Structure from a File**

**IPO**

* **Input**: Structure data from file
* **Process**: fread into struct variable
* **Output**: Display structure

#include <stdio.h>

struct Student {

char name[20];

int age;

};

int main() {

FILE \*fp;

struct Student s;

fp = fopen("student.dat", "rb");

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

fclose(fp);

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

return 0;

}

**Sample Output**

Name: John

Age: 20

## **9. Sort Names Stored in a File**

**IPO**

* **Input**: Names from file
* **Process**: Read into array, sort, write back
* **Output**: Names in sorted order

#include <stdio.h>

#include <string.h>

int main() {

FILE \*fp;

char names[10][20], temp[20];

int i=0, j, n;

fp = fopen("names.txt", "r");

while (fgets(names[i], sizeof(names[i]), fp)) {

names[i][strcspn(names[i], "\n")] = 0;

i++;

}

n = i;

fclose(fp);

for(i=0;i<n-1;i++)

for(j=i+1;j<n;j++)

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

strcpy(temp, names[i]);

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

strcpy(names[j], temp);

}

for(i=0;i<n;i++)

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

return 0;

}

**Sample Output**

Alex

John

Zara

## **10. Search for a Word in a File**

**IPO**

* **Input**: Word to search
* **Process**: Compare each word in file
* **Output**: Found / Not found

#include <stdio.h>

#include <string.h>

int main() {

FILE \*fp;

char word[20], temp[20];

int found = 0;

printf("Enter word to search: ");

scanf("%s", word);

fp = fopen("file1.txt", "r");

while (fscanf(fp, "%s", temp) == 1) {

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

found = 1;

break;

}

}

fclose(fp);

if (found) printf("Word found.\n");

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

return 0;

}

**Sample Output**

Enter word to search: Hello