**EXPERIMENT – 15**

15. Design a C program to organize the file using two level directory structure.

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#define MAX\_USERS 10

#define MAX\_FILES 10

#define MAX\_NAME 30

struct UserDirectory {

char userName[MAX\_NAME];

char fileNames[MAX\_FILES][MAX\_NAME];

int fileCount;

};

int findUser(struct UserDirectory users[], int userCount, char userName[]) {

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

if (strcmp(users[i].userName, userName) == 0) {

return i;

}

}

return -1;

}

void createUser(struct UserDirectory users[], int \*userCount) {

if (\*userCount >= MAX\_USERS) {

printf("Cannot create more user directories.\n");

return;

}

char name[MAX\_NAME];

printf("Enter new user name: ");

scanf("%s", name);

if (findUser(users, \*userCount, name) != -1) {

printf("User directory already exists.\n");

return;

}

strcpy(users[\*userCount].userName, name);

users[\*userCount].fileCount = 0;

(\*userCount)++;

printf("User directory '%s' created.\n", name);

}

void createFile(struct UserDirectory users[], int userCount) {

char name[MAX\_NAME];

printf("Enter user name: ");

scanf("%s", name);

int idx = findUser(users, userCount, name);

if (idx == -1) {

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

return;

}

if (users[idx].fileCount >= MAX\_FILES) {

printf("Cannot create more files for user '%s'.\n", name);

return;

}

char fname[MAX\_NAME];

printf("Enter file name to create: ");

scanf("%s", fname);

for (int i = 0; i < users[idx].fileCount; i++) {

if (strcmp(users[idx].fileNames[i], fname) == 0) {

printf("File already exists.\n");

return;

}

}

strcpy(users[idx].fileNames[users[idx].fileCount], fname);

users[idx].fileCount++;

printf("File '%s' created in user directory '%s'.\n", fname, name);

}

void deleteFile(struct UserDirectory users[], int userCount) {

char name[MAX\_NAME];

printf("Enter user name: ");

scanf("%s", name);

int idx = findUser(users, userCount, name);

if (idx == -1) {

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

return;

}

char fname[MAX\_NAME];

printf("Enter file name to delete: ");

scanf("%s", fname);

for (int i = 0; i < users[idx].fileCount; i++) {

if (strcmp(users[idx].fileNames[i], fname) == 0) {

for (int j = i; j < users[idx].fileCount - 1; j++) {

strcpy(users[idx].fileNames[j], users[idx].fileNames[j + 1]);

}

users[idx].fileCount--;

printf("File '%s' deleted from user '%s'.\n", fname, name);

return;

}

}

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

}

void displayUserFiles(struct UserDirectory users[], int userCount) {

char name[MAX\_NAME];

printf("Enter user name: ");

scanf("%s", name);

int idx = findUser(users, userCount, name);

if (idx == -1) {

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

return;

}

printf("Files in user '%s':\n", name);

if (users[idx].fileCount == 0) {

printf("No files.\n");

} else {

for (int i = 0; i < users[idx].fileCount; i++) {

printf("- %s\n", users[idx].fileNames[i]);

}

}

}

void displayAll(struct UserDirectory users[], int userCount) {

printf("\n--- All User Directories and Files ---\n");

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

printf("User: %s\n", users[i].userName);

if (users[i].fileCount == 0) {

printf(" No files.\n");

} else {

for (int j = 0; j < users[i].fileCount; j++) {

printf(" - %s\n", users[i].fileNames[j]);

}

}

}

}

int main() {

struct UserDirectory users[MAX\_USERS];

int userCount = 0;

int choice;

while (1) {

printf("\n--- Two-Level Directory Menu ---\n");

printf("1. Create User Directory\n");

printf("2. Create File\n");

printf("3. Delete File\n");

printf("4. Display Files of a User\n");

printf("5. Display All Users and Files\n");

printf("6. Exit\n");

printf("Enter choice: ");

scanf("%d", &choice);

switch (choice) {

case 1: createUser(users, &userCount); break;

case 2: createFile(users, userCount); break;

case 3: deleteFile(users, userCount); break;

case 4: displayUserFiles(users, userCount); break;

case 5: displayAll(users, userCount); break;

case 6: exit(0);

default: printf("Invalid choice.\n");

}

}

return 0;

}

SAMPLE OUTPUT:

--- Two-Level Directory Menu ---

1. Create User Directory

2. Create File

3. Delete File

4. Display Files of a User

5. Display All Users and Files

6. Exit

Enter choice: 1

Enter new user name: alice

User directory 'alice' created.