**Ex. No: 12**
**Date:** 1/4/25

# File Organization Technique- Single- and Two-level directory

**AIM:**

To implement File Organization Structures in C are

a. Single Level Directory

b. Two-Level Directory

c. Hierarchical Directory Structure

d. Directed Acyclic Graph Structure

## A. SINGLE LEVEL

**DIRECTORY ALGORITHM:**

1. Start
2. Declare the number, names and size of the directories and file names. 3. Get the values for the declared variables.
4. Display the files that are available in the directories.
5. Stop.

**PROGRAM:**

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

struct File {
   char name[20];
};

int main() {
   int n, i;
   struct File files[10];

   printf("Enter the number of files: ");
   scanf("%d", &n);

   if (n <= 0 || n > 10) {
      printf("Please enter a valid number of files (1–10).\n");
      return 1;
   }
```

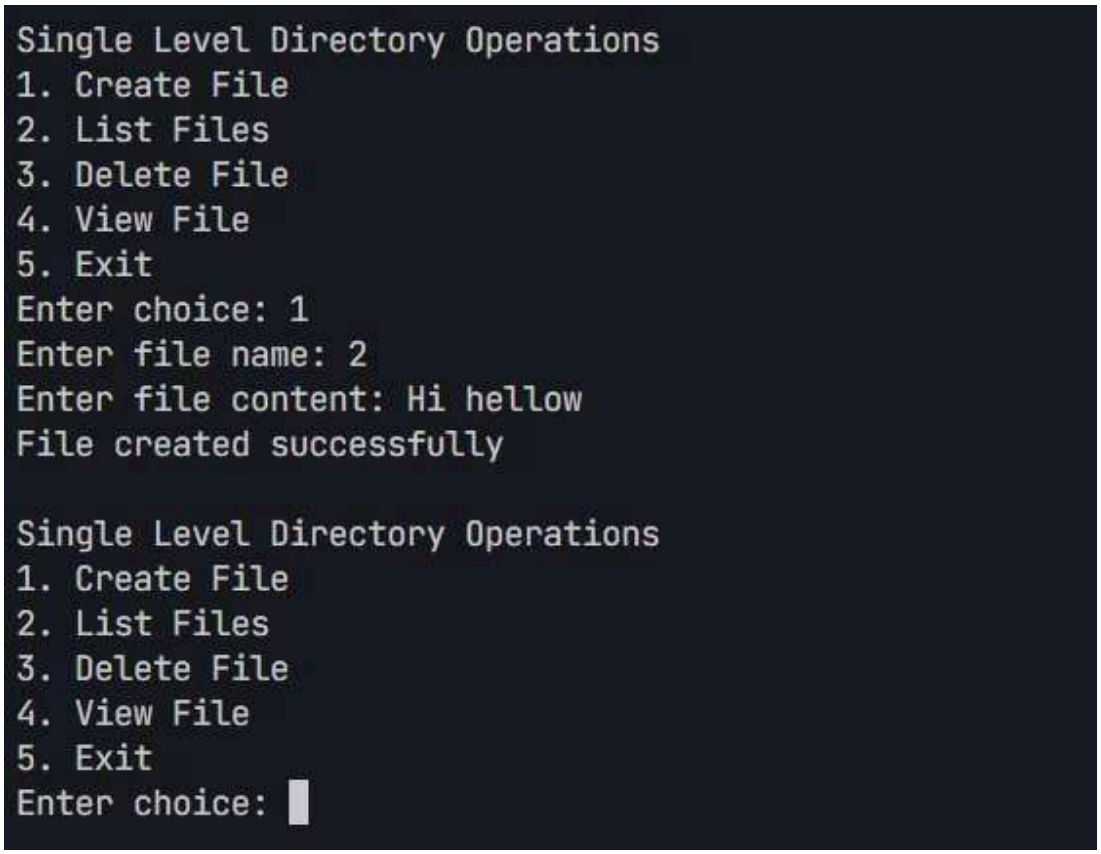
```
    for (i = 0; i < n; i++) {
        printf("Enter the file %d: ", i + 1);
        scanf("%s", files[i].name);
    }

    printf("\n\nRoot Directory\n");
    printf("|\n");

    for (i = 0; i < n; i++) {
        printf("|-- %s\n", files[i].name);
    }

    return 0;
}
```

OUTPUT:




**B. TWO-LEVEL DIRECTORY**

**STRUCTURE ALGORITHM:**

1. Start
2. Declare the number, names and size of the directories and subdirectories and file names.
3. Get the values for the declared variables.
4. Display the files that are available in the directories and subdirectories. 5. Stop.

**PROGRAM:**

```
#include <stdio.h> Implemented using C.
#include <string.h>

struct File {
   char name[20];
};

struct SubDirectory {
   char name[20];
   struct File
   files[10]; int
   fileCount;
};

struct Directory {
   char
   name[20];
   struct SubDirectory subDirs[10];
   int subDirCount;
};

int main() {
   struct Directory dir;
   int i, j;

   printf("Enter root directory name: ");
   scanf("%s", dir.name);

   printf("How many subdirectories in '%s'? ", dir.name);
   scanf("%d", &dir.subDirCount);

   for (i = 0; i < dir.subDirCount; i++) {
      printf("\nEnter name of subdirectory %d under '%s': ", i + 1, dir.name);
      scanf("%s", dir.subDirs[i].name);

      printf("How many files in '%s'? ", dir.subDirs[i].name);
      scanf("%d", &dir.subDirs[i].fileCount);

      for (j = 0; j < dir.subDirs[i].fileCount; j++) {
         printf("Enter file %d in '%s': ", j + 1, dir.subDirs[i].name);
         scanf("%s", dir.subDirs[i].files[j].name);
      }
   }


   printf("\nDirectory Structure:\n");
   printf("NULL\n");
   printf("|__%s\n", dir.name);

   for (i = 0; i < dir.subDirCount; i++) {
```

```
        printf("    |__%s\n", dir.subDirs[i].name);
        for (j = 0; j < dir.subDirs[i].fileCount; j++) {
            printf("        |__%s\n", dir.subDirs[i].files[j].name);
            }
    }

    return 0;
}
```

OUTPUT:

```
Single Level Directory Operations
1. Create File
2. List Files
3. Delete File
4. View File
5. Exit
Enter choice: 1
Enter file name: 2
Enter file content: Hi hellow
File created successfully

Single Level Directory Operations
1. Create File
2. List Files
3. Delete File
4. View File
5. Exit
Enter choice:
```

**RESULT:**

The File Organization Technique-Single and Two-Level Directory Program is Successfully Implemented