Ex. No.: 12
Date:

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>

#define MAX_FILES 5
#define MAX_USERS 3

// Single-Level Directory structure
void singleLevelDirectory() {
   // Array of files in the root directory
   char files[MAX_FILES][50] = {"file1.txt", "image1.png", "file2.txt", "document.pdf", "audio.mp3"};

   printf("\nSingle-Level Directory:\n");
   printf("Root Directory: \n");

   // Display files in the root directory
   for (int i = 0; i < MAX_FILES; i++) {
      printf("- %s\n", files[i]);
```

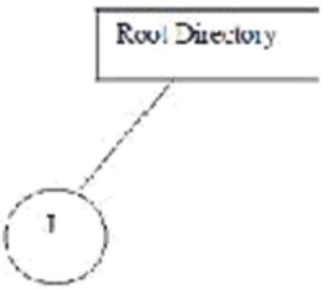
}

OUTPUT:

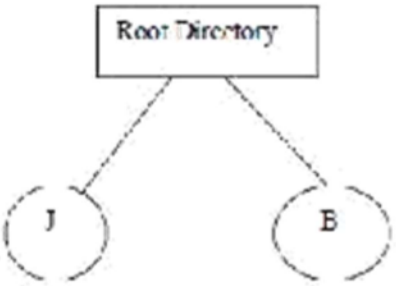
Enter the Number of files

2

Enter the file1 J



Enter the file2 B



}

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.

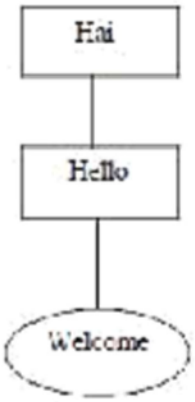
PROGRAM:

```
// Two-Level Directory structure
void twoLevelDirectory() {
   // Array of users and their respective files
   char users[MAX_USERS][50] = {"user1", "user2", "user3"};
   char userFiles[MAX_USERS][MAX_FILES][50] = {
      {"file1.txt", "image1.png", "doc1.pdf", "notes.txt", "audio.mp3"},
      {"file2.txt", "spreadsheet.xls", "image2.jpg", "report.docx", "presentation.pptx"},
      {"file3.txt", "music.mp3", "picture.png", "document.pdf", "game.exe"}
```

```
    };

    printf("\nTwo-Level Directory:\n");
    for (int i = 0; i < MAX_USERS; i++) {
        printf("\nUser: %s\n", users[i]);
        printf("Files:\n");

        // Display files for each user
        for (int j = 0; j < MAX_FILES; j++) {
            printf("- %s\n", userFiles[i][j]);
        }
    }
}

int main() {
    int choice;

    printf("Choose Directory Organization Technique:\n");
    printf("1. Single-Level Directory\n");
    printf("2. Two-Level Directory\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);

    switch(choice) {
        case 1:
            singleLevelDirectory();  // Display Single-Level Directory
            break;
        case 2:
            twoLevelDirectory();     // Display Two-Level Directory
            break;
        default:
            printf("Invalid choice! Please enter 1 or 2.\n");
            break;
    }

    return 0;
}
```

Sample Output:

Enter the name of dir/file(under null): Hai
How many users(for Hai):1
Enter name of dir/file(under Hai):Hello
How many files(for Hello):1
Enter name of dir/file(under Hello):welcome



Result:

The **File Organization Structures** were successfully implemented in **C**,
demonstrating the practical creation and management of various directory models including **Single Level**, **Two-Level**, **Hierarchical**, and **Directed Acyclic Graph (DAG)** structures,
highlighting the evolution from **simple file systems** to **complex, user-oriented and shared directory architectures**.