Ex. No.: 12
Date: 24-04-25

AIM: **File Organization Technique- Single and Two level directory**

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 <stdlib.h>
#include <graphics.h>
int main() {
    int gd = DETECT, gm, count, i, j, mid, div_x;
    char frame[10][20];
    int graph(&gd, &gm, "c:\\tc\\bgi");
    clear device();
    set bkdir (Cyan);
    puts("Enter the no of files");
    scanf("%d", &count);
```

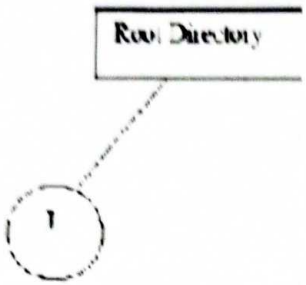
```
for (i=0; i<count ;i++) {
    clear device ();
    Set bkcolor ( GREEN);
    printf("Enter the file %d name", i+1);
    Scanf ("%s", frame [i]);
   Set fillstyle (1, MAGENTA);
   mid = 640/count : cir_x = mid/3;
   bar3d (270, 100, 370, 150, 0, 0);
    Settextstyle (2,0,4);
    Settextjustify (1,1);

    outtextxy (320, 125, "Root Directory");
    Setcolor (BLUE);
    for (j=0; j<=i, j++, cir_x+ =mid) {

            Line (320, 150, cir_x, 250);
            fillellipse (cir_x, 250, 30, 30);
            outtextxy ( cir_x, 250, frame [i]);
    }
}
```

OUTPUT:

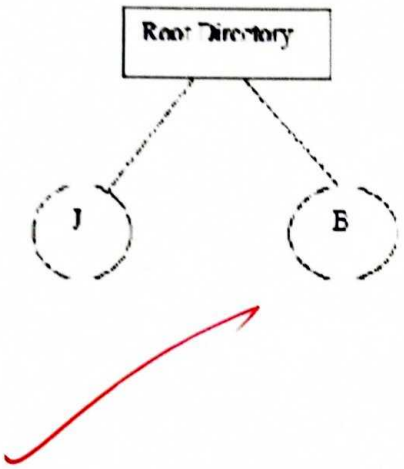
Enter the Number of files

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:

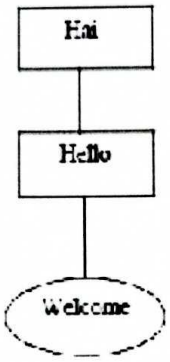
```
#Include <stdio.h>
#Include <graphics.h>
Struct tree element
{
  char name [20];
int x,y, ftype, lx, rx, nc, level; struct tree_element
* link [5] }; type def struct tree_element node;

void main() {
        int gd = DETECT , gm; node *root;
        root = NULL; clr scr();
        create (&root, 0, "null", 0, 630, 320);
        clr scr ();
        int graph (&gd , &gm; ".e : \\tc \\bgi");
        display (root);
         get ch ();
           close graph ();
}
```

```
reate(node **root, int lev, char &name, int x, int y)
{
  int i, gap;
{
(*root) = (node*) malloc (size of (node));
printf ("enter name of dir/ file( under %s): "; &name);
fflush (stdin)
gets ((*root) -> name);
if (lev == 0 || x == 1);
else
(*root) -> ftype = 2;
(*root) -> level = x;
(*root) -> y = 50 + x * 50;
(*root) -> x = x;
 for ((*root) -> ftype == 1)
{
  if (x == 0 || lev == 1)
  {
  if ((*root) -> level == 0)
   printf (" How many users");
else
   printf (" how many files");
   printf (" for %s) ; (*root) -> name);
}
else
  (*root -> n( = 0;
}
}
 display (node* root)
```

```
if (root != NULL)
{
  for (i=0; i<root -> nc; i++)
  {
    line (root -> x, root -> y, root -> x, root -> link[i] -> y)
}
 if (root -> x, root -> y, root -> link[i] -> y);
  triangle (root -> x, root -> y, root -> name);
   for (i=0; i<root -> nc; i++)
   {
     display (root -> link[i]);
   }
 }
}
```

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:

Thus the file organization technique single and two level directory and executed sucessfully.