OPERATING SYSTEM - CS23431

EXP 12

FILE ORGANISATION TECHNIQUE – SINGLE AND TWO LEVEL DIRECTORY

NAME: S.KUMARAN **ROLL NO: 230701159**

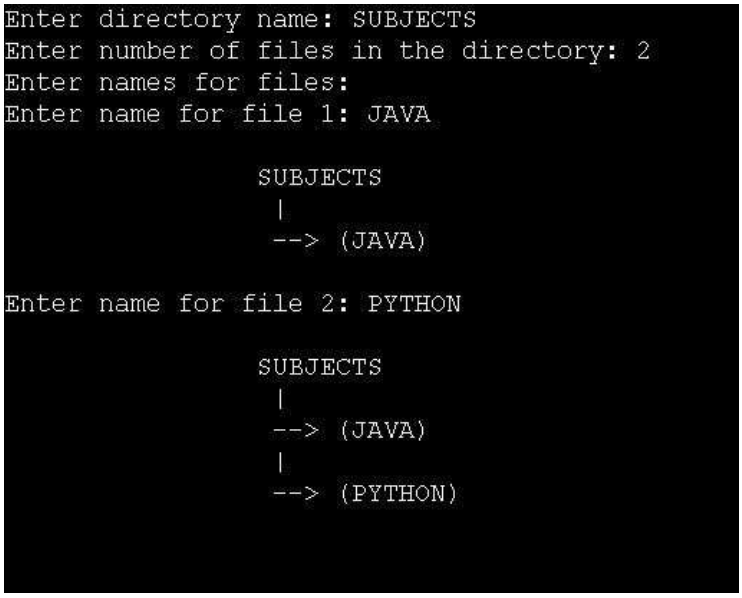
PROGRAM:

Single level directory:

```
#include <stdio.h>

struct directory {
   char dname[20];
   char fname[10][20];
   int f_count;
};

int main() {
   struct directory d;

   printf("Enter directory name: ");
   scanf("%s", d.dname);

   printf("Enter number of files in the directory: ");
   scanf("%d", &d.f_count);

   printf("Enter names for files:\n");

   for (int i = 0; i < d.f_count; i++) {
      printf("Enter name for file %d: ", i + 1);
      scanf("%s", d.fname[i]);

      printf("\n\t\t%s\n", d.dname);

      for (int j = 0; j <= i; j++) {
         printf("\t\t |\n");
         printf("\t\t --> (%s)\n", d.fname[j]);
      }
```

```
        printf("\n");
    }

    return 0;
}
```

OUTPUT:



Two level directory:

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

struct directory {
    char dname[20];
    char subnames[10][20];
    int sub_count;
};

int main() {
    struct directory d;      // Main directory
    struct directory sub[10]; // Subdirectories or sub-files

    // Input root directory name
```

```
printf("Enter the name of dir/file (under null): ");
scanf("%s", d.dname);

// Number of sub-items under root directory
printf("How many users (for %s): ", d.dname);
scanf("%d", &d.sub_count);

// Input each subdirectory or file under root directory
for (int i = 0; i < d.sub_count; i++) {
   printf("Enter the name of dir/file (under %s): ", d.dname);
   scanf("%s", d.subnames[i]);

   // Input number of sub-items under this subdirectory
   printf("How many users (for %s): ", d.subnames[i]);
   scanf("%d", &sub[i].sub_count);

   strcpy(sub[i].dname, d.subnames[i]);

   // Input names under each subdirectory
   for (int j = 0; j < sub[i].sub_count; j++) {
      printf("Enter name of dir/file (under %s): ", sub[i].dname);
      scanf("%s", sub[i].subnames[j]);
   }
}

// Printing the directory structure
printf("\nDirectory Structure:\n");
for (int i = 0; i < d.sub_count; i++) {
   for (int j = 0; j < sub[i].sub_count; j++) {
      printf("\t\t | %s |\n", d.dname);
      printf("\t\t |\n");
      printf("\t\t | %s |\n", sub[i].dname);
      printf("\t\t |\n");
      printf("\t\t ( %s )\n", sub[i].subnames[j]);
      printf("\n");
```

```
    }
  }

  return 0;
}
```

OUTPUT: