WALCHAND COLLEGE OF ENGINEERING SANGLI

Department of Information Technology

UNIX OPERATING SYSTEM LAB (3IT 371)

Year of Studentship: 2021-

22 Term: Semester-2

Class: T.Y. IT

Name	**PRN. No**
Aditi Sudhir Ghate	**2020BTEIT00044**

Chapter 3 : File system Internals

3.a - Write the program to show file statistics using the stat system call. Take the filename / directory name from user including path.

Objectives :

1. To learn about File system Internals.

Theory :

Name:

    stat, fstat, lstat - get file status

Syntax:

```
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>

int stat(const char *path, struct stat *buf);
int fstat(int fd, struct stat *buf);
int lstat(const char *path, struct stat *buf);
```

Description:

These functions return information about a file. No permissions are required on the file itself, but-in the case of stat() and lstat() - execute (search) permission is required on all of the directories in *path* that lead to the file.

stat() stats the file pointed to by *path* and fills in *buf*.

lstat() is identical to stat(), except that if *path* is a symbolic link, then the link itself is stat-ed, not the file that it refers to.

fstat() is identical to stat(), except that the file to be stat-ed is specified by the file descriptor *fd*.

All of these system calls return a *stat* structure, which contains the following fields :

struct stat {

dev_t   st_dev;   /* ID of device containing file */

ino_t   st_ino;   /* inode number */

 mode_t st_mode;   /* protection */

nlink_t st_nlink; /* number of hard links */

uid_t   st_uid;   /* user ID of owner */ gid_t

         st_gid;   /* group ID of owner */

dev_t   st_rdev;   /* device ID (if special file) */

off_t   st_size; /* total size, in bytes */

blksize_t st_blksize; /* blocksize for file system I/O */

blkcnt_t st_blocks; /* number of 512B blocks allocated */

time_t   st_atime; /* time of last access */

time_t   st_mtime; /* time of last modification */

time_t   st_ctime;   /* time of last status change */

 };

Data Dictionary :

Sr Number	Variable/Function	Datatype	Use
1	fileStat	struct stat	Store information about files.

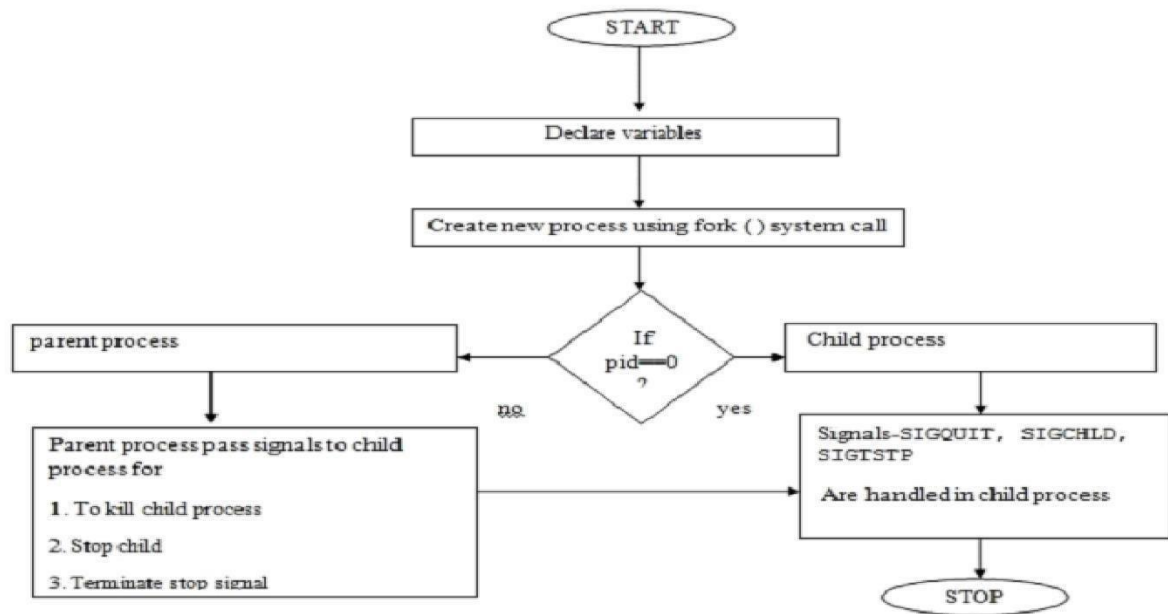**Flowchart :**

Fig: 3.1 Flowchart

Program :

```c
#include<stdio.h>
#include<unistd.h>
#include<sys/types.h>
#include<sys/stat.h>
int main(int argc, char **argv)
{
struct stat fileStat; stat("/home/aditi/ADnOR/Assignments/3A/readme",&fileStat);
if(stat("/home/aditi/ADnOR/Assignments/3A/readme",&fileStat) < 0)
{
printf("Failed\n");
return 1;
}
printf("---------------------------\n");
printf("File Size: \t\t%ld bytes\n",(long)fileStat.st_size);
printf("Number of Links: \t%ld\n",(long)fileStat.st_nlink); printf("File inode: \t\t%ld\n",(long)fileStat.st_ino);
printf("File Permissions: \t");

printf( (S_ISDIR(fileStat.st_mode)) ? "d" : "-");
printf( (fileStat.st_mode & S_IRUSR) ? "r" : "-");
printf( (fileStat.st_mode & S_IWUSR) ? "w" : "-");
printf( (fileStat.st_mode & S_IXUSR) ? "x" : "-");
printf( (fileStat.st_mode & S_IRGRP) ? "r" : "-");
printf( (fileStat.st_mode & S_IWGRP) ? "w" : "-");
printf( (fileStat.st_mode & S_IXGRP) ? "x" : "-");
printf( (fileStat.st_mode & S_IROTH) ? "r" : "-");
printf( (fileStat.st_mode & S_IWOTH) ? "w" : "- ");
printf( (fileStat.st_mode & S_IXOTH) ? "x" : "- ");
printf("\n\n");
printf("The file %s a symbolic link\n", (S_ISLNK(fileStat.st_mode)) ? "is" : "is not");
return 0;
}
```

Output :

```
aditi@aditi-Lenovo-ideapad-330S-14IKB-U:~/ADnOR/Assignments/3A$ gcc 3A.c
aditi@aditi-Lenovo-ideapad-330S-14IKB-U:~/ADnOR/Assignments/3A$ ./a.out
---------------------------
File Size:              12 bytes
Number of Links:        1
File inode:             4587524
File Permissions:       -rw-rw-r- -

The file is not a symbolic link
```

Conclusion :

- Stats of file like file size,links, permissions, inode number and type of link can be retrieved using stat() and stored in a structure.

References :

[1] https://www.lix.polytechnique.fr/~liberti/public/computing/prog/c/C/FUNCTIONS/stat.h