

Nama : Aulia Putri R

NIM : L200180156

Kelas : D

MODUL 8

The image shows a C program in a code editor and its execution output in a terminal. The code is a program named `stat.c` that uses the `stat` system call to retrieve file metadata. It includes headers for `stdio.h`, `sys/stat.h`, `stdlib.h`, and `time.h`. The `main` function takes command-line arguments and prints usage information if there is only one argument. Otherwise, it calls `stat` on the provided file and prints various file statistics, including user/group IDs, block size, blocks allocated, inode number, last accessed/modified times, file size, number of links, permission bits, and file type (regular or directory).

```
#include <stdio.h>
#include <sys/stat.h>
#include <stdlib.h>
#include <time.h>

int main(int argc, char*argv[]) {
    struct stat
    file;
    int n;
    if (argc != 1)
    {
        printf("Usage: ./a.out <filename>\n");
        exit(-1);
    }
    if ((n = stat(argv[1], &file)) == -1)
    {
        perror(argv[1]);
        exit(-1);
    }
    printf("User id : %d\n", file.st_uid);
    printf("Group id : %d\n", file.st_gid);
    printf("Block size : %ld\n", file.st_blksize);
    printf("Blocks allocated : %ld\n", file.st_blocks);
    printf("Inode no. : %ld\n", file.st_ino);
    printf("Last accessed : %s", ctime(&(file.st_atime)));
    printf("Last modified : %s", ctime(&(file.st_mtime)));
    printf("File size : %ld bytes\n", file.st_size);
    printf("No. of links : %ld\n", file.st_nlink);
    printf("Permission : ");
    printf((S_ISDIR(file.st_mode)) ? "d" : "-");
    printf((file.st_mode & S_IRUSR) ? "r" : "-");
    printf((file.st_mode & S_IWUSR) ? "w" : "-");
    printf((file.st_mode & S_IXUSR) ? "x" : "-");
    printf((file.st_mode & S_IRGRP) ? "r" : "-");
    printf((file.st_mode & S_IWGRP) ? "w" : "-");
    printf((file.st_mode & S_IXGRP) ? "x" : "-");
    printf((file.st_mode & S_IROTH) ? "r" : "-");
    printf((file.st_mode & S_IWOTH) ? "w" : "-");
    printf((file.st_mode & S_IXOTH) ? "x" : "-");
    printf("\n");
    if(file.st_mode & S_IFREG)
        printf("File type : Regular\n");
    if(file.st_mode & S_IFDIR)
        printf("File type : Directory\n");
}
```

The terminal output shows the program being compiled and executed. The first execution shows the usage message. The second execution shows the file type for `/bin/ls`. The third execution shows the file type for `/`.

```
dhlnas@dhlnas:~/Documents$ gcc stat.c
dhlnas@dhlnas:~/Documents$ ./a.out
Usage: ./a.out <filename>
dhlnas@dhlnas:~/Documents$ ./a.out /bin/ls
Usage: ./a.out <filename>
dhlnas@dhlnas:~/Documents$
```

Activities Text Editor Nov 27 13:32 •
dirlist.c
~/Downloads/Module

```
#include <stdio.h>
#include <dirent.h>
#include <stdlib.h>
int main(int argc, char*argv[]) {
    struct dirent *dptr;
    DIR *dname;

    if (argc != 2)
    {
        printf("Usage: ./a.out <dirname>\n");
        exit(-1);
    }
    if ((dname = opendir(argv[1])) == NULL)
    {
        perror(argv[1]);
        exit(-1);
    }
    while (dptr=readdir(dname)) {
        printf("%s\n", dptr->d_name);
    }
    closedir(dname);
}
```

Save

C Tab Width: 8 Ln 4, Col 34 INS

```
dhlnas@dhlnas:~/Documents$ gcc dirlist.c
dhlnas@dhlnas:~/Documents$ ls
a.out dirlist.c 'Module' sdf.c stat.c
dhlnas@dhlnas:~/Documents$ ./a.out
Usage: ./a.out <dirname>
dhlnas@dhlnas:~/Documents$ ./a.out /bin/ls ls
Usage: ./a.out <dirname>
dhlnas@dhlnas:~/Documents$
```