Lab-4 OS

Name :Sagnik Chatterjee

Sec :B

SEM:5

RollNo :61

Reg :180905478

Q1

Write a program to find the inode number of an existing file in a directory. Take the input as a filename and print the inode number of the file

Code :

/\*

AUTHOR :SAGNIK CHATTERJEE

DATE : 11 DEC,2020

USAGE : ./q1 q1file

where q1file is the input file

\*/

//to print the inode number of the file

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

int main(int argc ,char \*\*argv){

if(argc!=2){

printf("[ERROR] Usage : %s <file> \n",argv[0]);

exit(1);

}

char buffer[100];

bzero(buffer,sizeof(buffer));

strcat(buffer,"ls ");

strcat(buffer,"-i ");

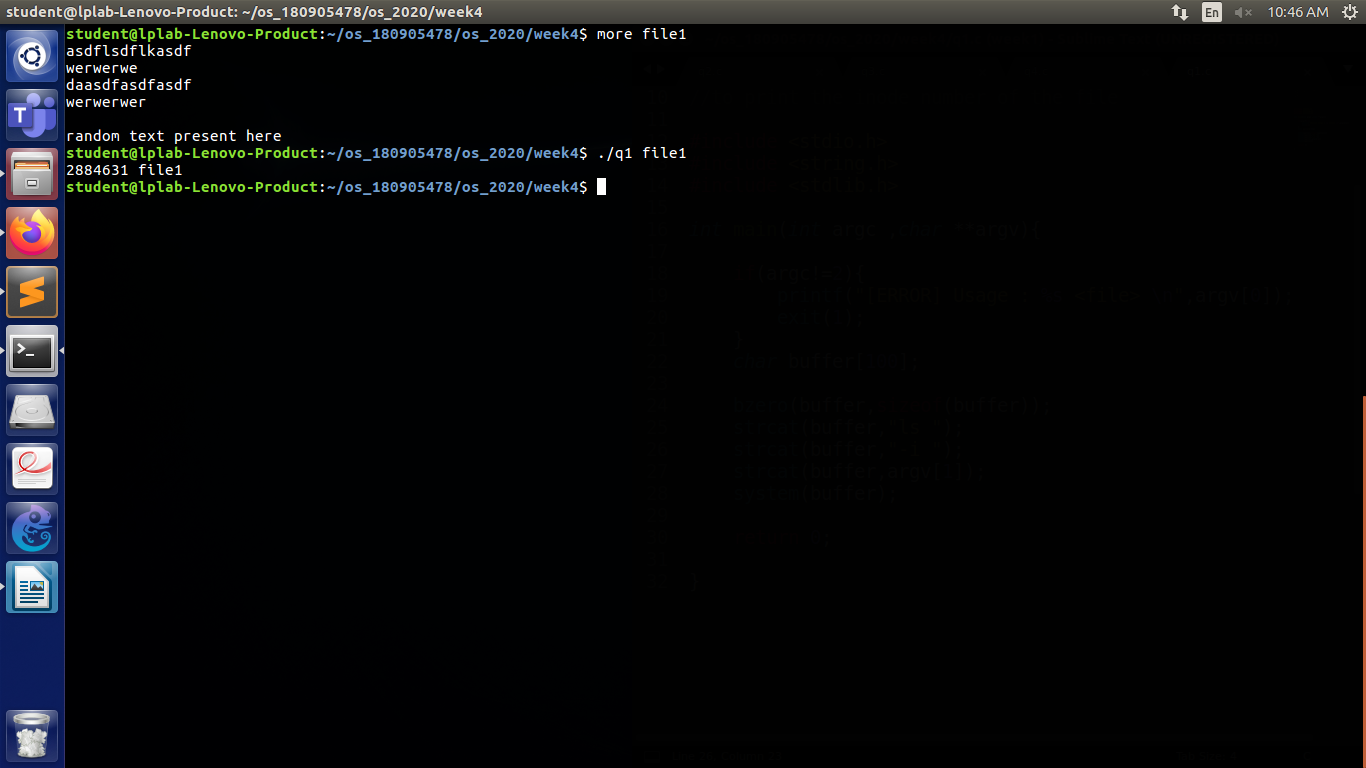
strcat(buffer,argv[1]);

system(buffer);

return 0;

}

Screenshot :



Q2

Write a program to print out the complete stat structure of a file.

Code :

/\*

AUTHOR :SAGNIK CHATTERJEE

DATE : 11 DEC,2020

USAGE : ./q2 file

where file is the input file

\*/

//to print the complete stat structure of the file

#include <stdio.h>

#include <stdlib.h>

#include <sys/types.h>

#include <time.h>

#include <sys/stat.h>

#include <unistd.h>

int main(int argc ,char \*\*argv){

struct stat file\_stats;

if(argc!=2){

printf("[ERROR] Usage : %s <filename>",argv[0]);

exit(1);

}

if((stat(argv[1],&file\_stats))==-1){

printf("[ERROR] fstat error \n");

exit(1);

}

print("[STATUS] File Reports :\n");

printf(" Filename: %s\n", argv[1]);

printf(" Device: %lld\n", file\_stats.st\_dev);

printf(" Inode: %ld\n", file\_stats.st\_ino);

printf(" Time of last access: %ld : %s", file\_stats.st\_atime, ctime(&file\_stats.st\_atime));

printf(" Time of last modification: %ld : %s", file\_stats.st\_mtime, ctime(&file\_stats.st\_mtime));

printf(" Time of last change: %ld : %s", file\_stats.st\_ctime, ctime(&file\_stats.st\_ctime));

printf(" Protection: %o\n", file\_stats.st\_mode);

printf(" Number of hard links: %d\n", file\_stats.st\_nlink);

printf(" User ID of owner: %d\n", file\_stats.st\_uid);

printf(" Group ID of owner: %d\n", file\_stats.st\_gid);

printf(" Device type (if inode device): %lld\n", file\_stats.st\_rdev);

printf(" Total size, in bytes: %ld\n", file\_stats.st\_size);

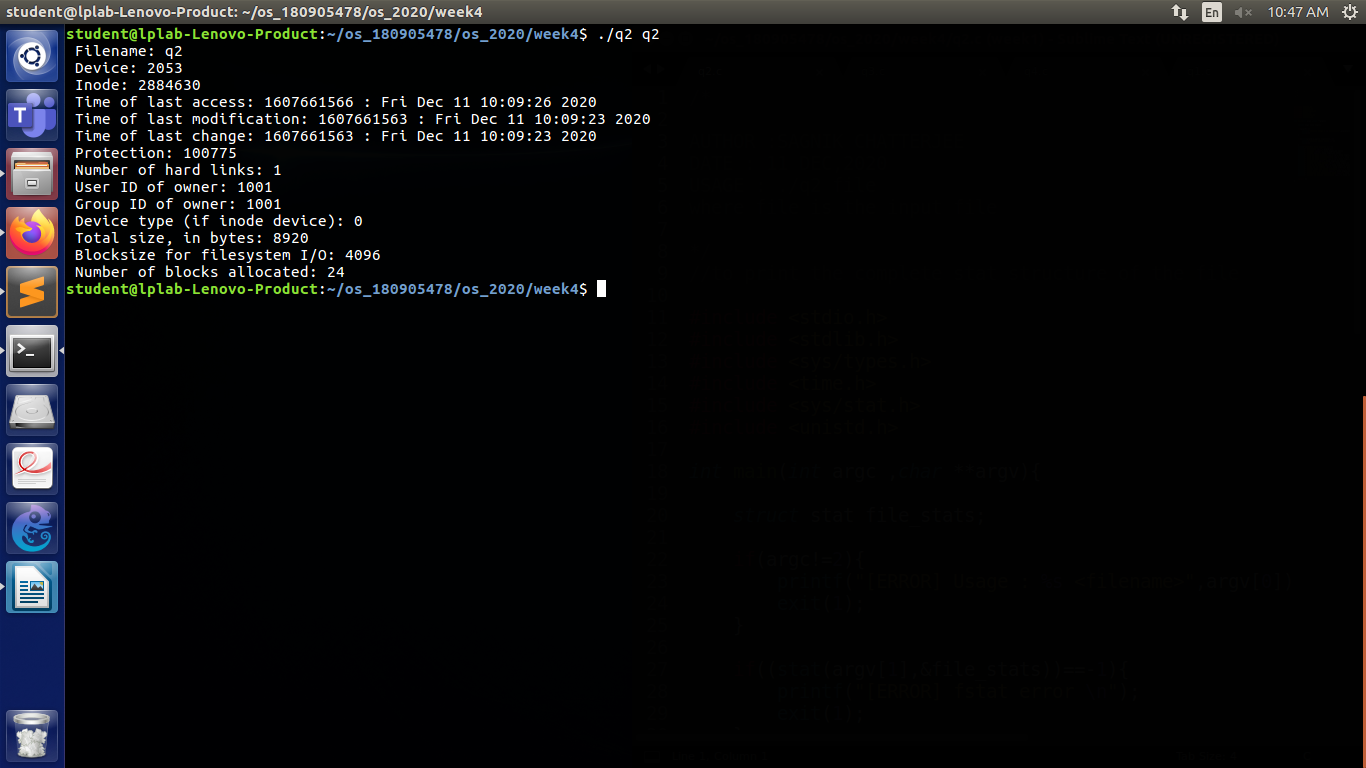
printf(" Blocksize for filesystem I/O: %ld\n", file\_stats.st\_blksize);

printf(" Number of blocks allocated: %ld\n", file\_stats.st\_blocks);

return 0;

}

Screenshot :



Q3

Write a program to create a new hard link to an existing file and unlink the same. Accept the old path as input and print the newpath.

Code:

#include <time.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <unistd.h>

#include <sys/types.h>

#include <string.h>

int main(int argv, char \*argc[]) {

char command[50] = "ls -il";

printf("old path -> %s\n", argc[1]);

link(argc[1], "link");

system(command);

printf("\n");

unlink(argc[1]);

system(command);

}

Screenshot:

Q4

Write a program to create a new soft link to an existing file and unlink the same. Accept the old path as input and print the newpath.

Code:

#include <time.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <unistd.h>

#include <sys/types.h>

#include <string.h>

int main(int argv, char \*argc[]) {

char command[50] = "ls -il";

printf("old path -> %s\n", argc[1]);

symlink(argc[1], "link");

system(command);

printf("\n");

unlink(argc[1]);

system(command);

}

Screenshot :

