Lab 5 OS

Utkarsh Bajaj, Roll No. 53, Reg No. 180905460

Ans 1) Program:

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <sys/wait.h>

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

{

char file[100];

struct stat sb;

int ret;

printf("Enter the file name\n");

scanf("%s", file);

ret = stat(file, &sb);

if(ret){

perror("stat");

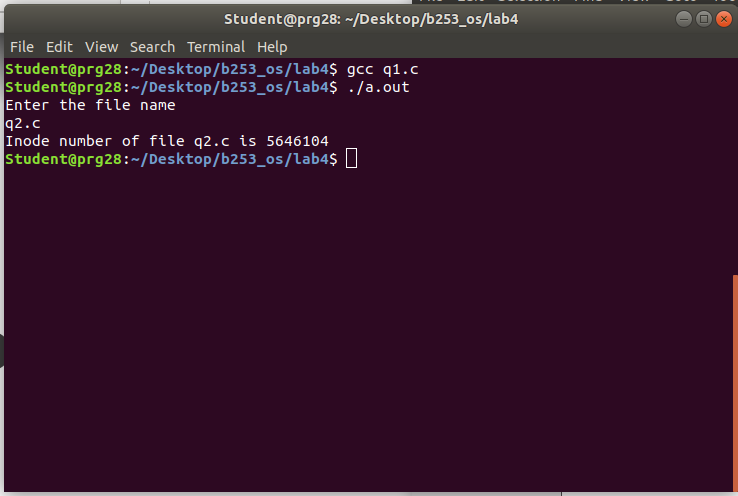
exit(1);

}

printf("Inode number of file %s is %ld\n", file, sb.st\_ino);

return 0;

}



Ans 2) Program :

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <sys/wait.h>

#include <time.h>

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

{

char file[100];

struct stat sb;

int ret;

printf("Enter the file name\n");

scanf("%s", file);

ret = stat(file, &sb);

if(ret){

perror("stat");

exit(1);

}

printf("\nID of device: %ld",sb.st\_dev);

printf("\nInode no.: %ld",sb.st\_ino);

printf("\nPermissions: %o",sb.st\_mode);

printf("\nNo. of hard links: %ld",sb.st\_nlink);

printf("\nUser ID of owner: %d",sb.st\_uid);

printf("\ngroup ID of owner: %d",sb.st\_gid);

printf("\ndevice ID: %ld",sb.st\_rdev);

printf("\nTotal size: %ld",sb.st\_size);

printf("\nBlocksize: %ld",sb.st\_blksize);

printf("\nNo. of blocks: %ld",sb.st\_blocks);

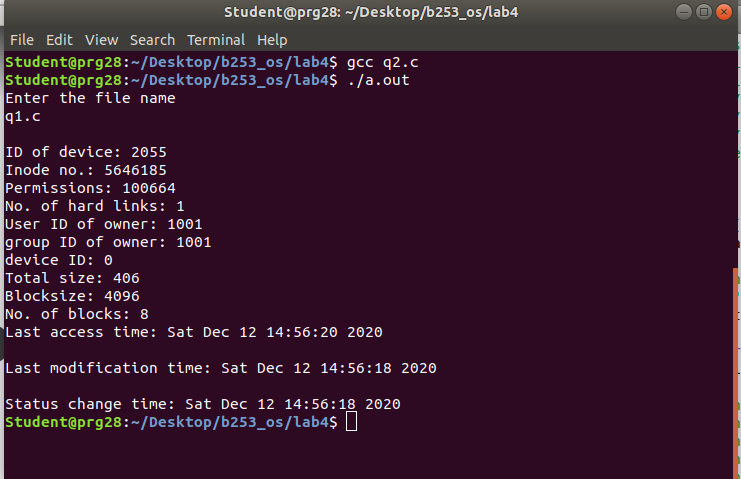
printf("\nLast access time: %s",ctime(&sb.st\_atime));

printf("\nLast modification time: %s",ctime(&sb.st\_mtime));

printf("\nStatus change time: %s",ctime(&sb.st\_ctime));

exit(0);

}



Ans 3)   
Program:

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <sys/wait.h>

#include <time.h>

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

struct stat sb;

int ret;

puts("Before linking");

system("ls -il test.\*");

char path[100];

printf("Enter the pathname\n");

scanf("%s", path);

ret = link("test.txt", path);

if(ret){

printf("Error\n");

exit(1);

} else {

printf("After Link\n");

system("ls -il test.\*");

system("ls -il ./new/test2.txt");

}

ret = unlink("test.txt");

if(ret){

perror("Unlink error");

exit(1);

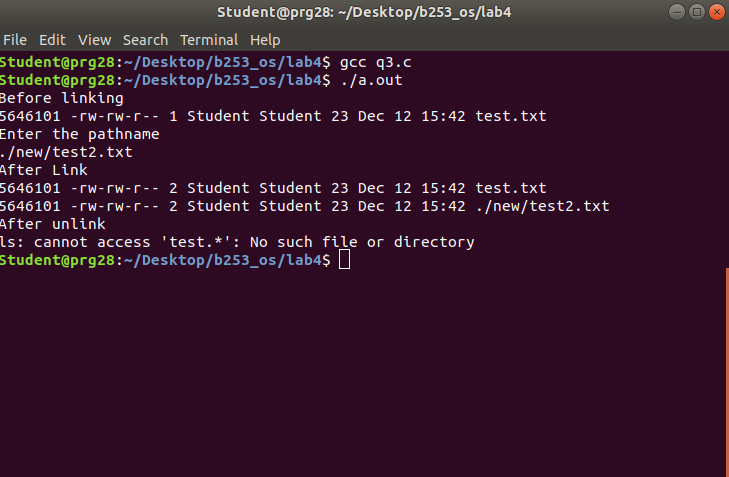
} else {

printf("After unlink\n");

system("ls -il test.\*");

}

}



Ans 4)

Program :

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <sys/wait.h>

#include <time.h>

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

struct stat sb;

int ret;

puts("Before linking");

system("ls -il test.\*");

char path[100];

printf("Enter the pathname\n");

scanf("%s", path);

ret = symlink("test.txt", path);

if(ret){

printf("Error\n");

exit(1);

} else {

printf("After Link\n");

system("ls -il test.\*");

system("ls -il ./new/test2.txt");

}

ret = unlink("test.txt");

if(ret){

perror("Unlink error");

exit(1);

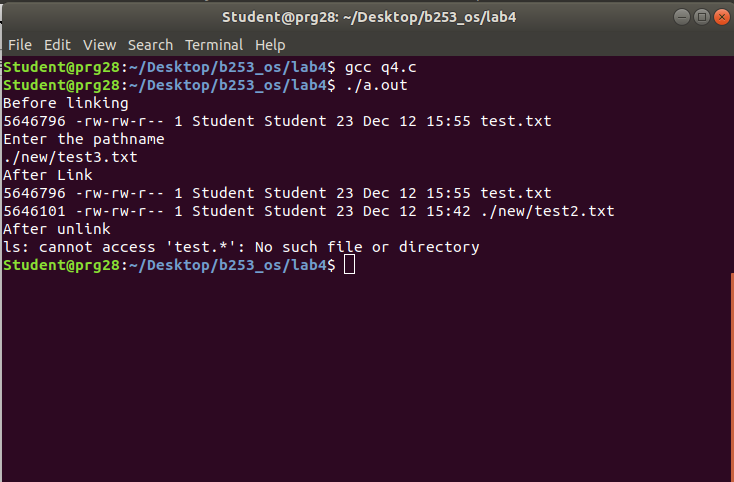
} else {

printf("After unlink\n");

system("ls -il test.\*");

}

}



After unlink:

