

CERTIFICATE

Name of the Lab : LINUX PROGRAMMING

Name of the Student : THOTA NAGABABU

Student Regd. No. : 18BQ1A05K3

CLASS : III B.TECH. I SEM CSE – D

GIT HUB LINK:

<https://github.com/nagababuthota984/5K3-OS-LAB>

INDEX

S. No.	Name of the Experiment	LAST DATE	PAGE NO
2	Write a C program that makes a copy of a file using standard I/O, and system calls	DEC 22	

EXPERIMENT NO: 2

AIM : To write a C program that makes a copy of a file using standard I/O, and system calls

DESCRIPTION :

Basically in Unix, there is a command to copy contents of file from one file to other, called **cp**. This command directly copies the contents and now to make this task work using a C program:-

special functions we need:-

perror()

exit()

open()

constants we need are:-

O_RDONLY

O_WRONLY

O_CREAT etc.

LIBRARIES USED:

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

```
#include <unistd.h>
```

```
#include <fcntl.h>
```

```
#include <stdlib.h>
```

These are the header files included libraries to execute different system calls required to write this program.

SYNTAX:

```
int main(int , char ){} 
```

open(): It is present in unistd library.

open() returns a file descriptor, a small, non-negative integer for use in subsequent system calls **read()**, **write()**, **lseek()**, **fcntl()**, etc .The parameter *flags* must include one of the

following *access modes*: **O_RDONLY**, **O_WRONLY**, or **O_RDWR**. These request opening the file read-only, write-only, or read/write, respectively.

int open(const char *pathname, int flags);

int open(const char *pathname, int flags, mode_t mode);

perror(): It is present in stdio library.

The `perror()` function produces a message on standard error describing the last error encountered during a call to a system or library function.

perror("Error message");

exit(): It is present in stdlib library

cause normal process termination.

void exit(int status);

close() : It is present in unistd library.

The `close()` function shall deallocate the file descriptor indicated by *fdes*. To deallocate means to make the file descriptor available for return by subsequent calls to `open()` or other functions that allocate file descriptors

int close(int fides);

PROGRAM:

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

```
#include <unistd.h>
```

```
#include <fcntl.h>
```

```
#include <stdlib.h>
```

```
int main(int argc, char *argv[])
```

```
{
```

```
int f1, f2;
```

```
char buff[50];
```

```
long int n;
```

```
if(((f1 = open(argv[1], O_RDONLY)) == -1 || ((f2=open(argv[2], O_CREAT |  
O_WRONLY | O_TRUNC, 0700))== 1)))
```

```
{
```

```
perror("problem in file");
```

```
exit(1);
```

```
}
```

```
while((n=read(f1, buff, 50))>0)
```

```
if(write(f2, buff, n)!=n)
```

```
{
```

```
perror("problem in writing");
```

```
exit(3);
```

```
}
```

```
if(n== -1)
```

```
{
```

```
perror("problem in reading");
```

```
exit(2);
```

```
}
```

```
close(f2);
```

```
exit(0);
```

```
}
```

OUTPUT:

After executing the cp command, we can observe the contents of a file are copied to the destination file.

OUTPUT SCREEN SHOTS:

```
3-cse-d@Lab-04-24:~/18BQ1A05K3/UNIX LAB$ gcc copy.c
3-cse-d@Lab-04-24:~/18BQ1A05K3/UNIX LAB$ ./a.out copy.c dest.txt
3-cse-d@Lab-04-24:~/18BQ1A05K3/UNIX LAB$ cat dest.txt
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{
    int f1, f2;
    char buff[50];
    long int n;

    if(((f1 = open(argv[1], O_RDONLY)) == -1 || ((f2=open(argv[2], O_CREAT | O_WRONLY | O_TRUNC, 0700))== 1)))
    {
        perror("problem in file");
        exit(1);
    }

    while((n=read(f1, buff, 50))>0)
    {
        if(write(f2, buff, n)!=n)
        {
            perror("problem in writing");
            exit(3);
        }
        if(n==1)
        {
            perror("problem in reading");
            exit(2);
        }
    }

    close(f2);
    exit(0);
}
```

We can observe that the contents from file 'filecopy.c' have been copied to the destination file 'des.txt'.



VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY

Permanently Affiliated to JNTU Kakinada, Approved by AICTE

Accredited by NAAC with 'A' Grade, ISO 9001:2008 Certified

Nambur, Pedakakani (M), Guntur (Dt) - 522508

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.Tech Program is Accredited by NBA