

## Kernel API's Implementation

### cp Command

Program:

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <errno.h>
#include <sys/types.h>
#include <unistd.h>
#define BUF_SIZE 8192
int main(int argc, char* argv[]) {
    int input_fd, output_fd;
    ssize_t ret_in, ret_out;
    char buffer[BUF_SIZE];
    if(argc != 3){
        printf ("Usage: cp file1 file2");
        return 1;
    }
    input_fd = open (argv [1], O_RDONLY);
    if (input_fd == -1) {
        perror ("open");
        return 2;
    }
    output_fd = open(argv[2], O_WRONLY | O_CREAT, 0644);
    if(output_fd == -1){
        perror("open");
        return 3;
    }
    while((ret_in = read (input_fd, &buffer, BUF_SIZE)) > 0){
        ret_out = write (output_fd, &buffer, (ssize_t) ret_in);
        if(ret_out != ret_in){
            perror("write");
            return 4;
        }
    }
    close (input_fd);
    close (output_fd);
    return (EXIT_SUCCESS);
}
```

Output:

nitin@nitin-VirtualBox:~\$ ./tt f.txt d.txt

nitin@nitin-VirtualBox:~\$ cat f.txt

hello

world

```
nitin@nitin-VirtualBox:~$ cat d.txt
hello
nitin@nitin-VirtualBox:~$ cc nitin.c -o tt
nitin@nitin-VirtualBox:~$ ./tt f.txt d.txt
nitin@nitin-VirtualBox:~$ cat d.txt
hello
world
```

### **mv Command**

Program:

```
#include<sys/types.h>
#include<sys/stat.h>
#include<stdio.h>
#include<fcntl.h>
main( int argc,char *argv[] )
{
    int i,fd1,fd2;
    char *file1,*file2,buf[2];
    file1=argv[1];
    file2=argv[2];
    printf("file1=%s file2=%s",file1,file2);
    fd1=open(file1,O_RDONLY,0777);
    fd2=creat(file2,0777);
    while(i=read(fd1,buf,1)>0)
    write(fd2,buf,1);
    remove(file1);
    close(fd1);
    close(fd2);
}
```

Output:

```
nitin@nitin-VirtualBox:~$ ls
Desktop  Downloads  examples.desktop  Music  nitin.c  Public  tt
Documents  d.txt    f.txt      mv.c  Pictures  Templates  Videos
nitin@nitin-VirtualBox:~$ cat f.txt
hello World
nitin@nitin-VirtualBox:~$ cat d.txt
nitin@nitin-VirtualBox:~$ cc mv.c -o tt
nitin@nitin-VirtualBox:~$ ./tt f.txt d.txt
file1=f.txt file2=d.txtnitin@nitin-VirtualBox:~$ ls
Desktop  Downloads  examples.desktop  mv.c  Pictures  Templates  Videos
Documents  d.txt    Music      nitin.c  Public  tt
nitin@nitin-VirtualBox:~$ cat d.txt
hello World
```

**grep Command:**

Program:

```
#include<string.h>
#include<fcntl.h>
#include<unistd.h>
#include<stdio.h>
#include<stdlib.h>
#include<sys/uio.h>
int main(int argc, char *argv[])
{
    int des, fhandle, i,k, arg_ind=2; int j;
    int linecount;
    int lineflag;
    char buf,line[160];
    if (argc < 4) // To check the syntax
    {
        printf( "\n Error... Correct Syntax is : grep \"pattern\" \"filename\"\\n");
        exit(0);
    }
    lineflag=0;
    des = open(argv[3],O_RDONLY);
    linecount=0;
    do
    {
        i=0;
        do
        {
            fhandle=read(des,&buf,1);
            line[i++]=buf;
        } while(( fhandle!= 0) && (buf != '\\n'));
        line[i]='\\0';
        linecount++;
        i=0;
        while(line[i] != '\\0')
        {
            if(line[i] == argv[2][0])
            {
                k=1;
                i++;
                while((k < strlen(argv[2])) && (line[i] == argv[2][k]))
                {
                    i++; k++;
                }
            }
        }
    }
```

```

if(k == strlen(argv[2]))
printf("%s",line);
}
i++;
}
}while( fhandle!= 0);
close(des);
arg_ind++;
printf( "\n");
}

```

Output:

```

nitin@nitin-VirtualBox:~$ cat d.txt
hello World
Operating System
System calls
nitin@nitin-VirtualBox:~$ cc grep.c -o tt
nitin@nitin-VirtualBox:~$ ./tt grep hello d.txt
hello World

```

### ls command

Program:

```

#include<stdio.h>
#include<dirent.h>
main()
{
char dirname[10];
DIR*p;
struct dirent *d;
printf("Enter directory name\n");
scanf("%s",dirname);
p=opendir(dirname);
if(p==NULL)
{
perror("Cannot find directory");
exit(-1);
}
while(d=readdir(p))
printf("%s\n",d->d_name);
}

```

Output:

```

Enter directory name
f:
$RECYCLE.BIN
Android Studio

```

en\_microsoft\_office\_2013\_32bit\_dvd.iso  
New folder  
NNT  
oracle  
PYS  
SOFTWARES  
System Volume Information  
TDM-GCC-32

### head command:

Program:

```
#include <stdio.h>
int main(int argc, char * argv[])
{
    FILE * fp;          // file pointer
    char * line = NULL;
    int len = 0;
    int cnt = 0;
    fp = fopen(argv[1], "r");
    if ( fp == NULL )
    {
        printf("\n%s file can not be opened !!!\n", argv[1]);
        return 1;
    }
    while (getline(&line, &len, fp) != -1)
    {
        cnt++;

        if ( cnt > atoi(argv[2]) )
            break;

        printf("%s", line); fflush(stdout);
    }
    fclose(fp);
    return 0;
}
```

Output:

```
nitin@nitin-VirtualBox:~$ cat d.txt
hello World
Operating System
System calls
n
5
6
```

9

10

10

200

nitin@nitin-VirtualBox:~\$ cc head.c -o tt

head.c: In function 'main':

head.c:19:20: warning: implicit declaration of function 'atoi' [-Wimplicit-function-declaration]

if ( cnt > atoi(argv[2]) )

^~~~

nitin@nitin-VirtualBox:~\$ ./tt d.txt 5

hello World

Operating System

System calls

n

5