|  |  |
| --- | --- |
| **Ex.No:3** | C programs to simulate UNIX commands like cp, ls, grep. |

**AIM:**

To write C programs to simulate UNIX commands like cp, ls, grep.

# Program for simulation of cp unix commands ALGORITHM:

STEP1: Start the program

STEP 2:Declare the variables ch, \*fp, sc=0 STEP3: Open the file in read mode

STEP 4: Get the character

STEP 5: If ch== “ “ then increment sc value by one STEP 6: Print no of spaces

STEP 7:Close the file

**PROGRAM:**

#include<fcntl.h> #include<unistd.h> #include<stdio.h> main(int argc,char \*argv[])

{

FILE \*fp; char ch; int sc=0;

fp=fopen(argv[1],"r"); if(fp==NULL)

printf("unable to open a file",argv[1]); else

{

while(!feof(fp))

{

ch=fgetc(fp); if(ch==' ') sc++;

}

printf("no of spaces %d",sc); printf("\n");

fclose(fp);

}

}

# PROGRAM FOR SIMULATION OF LS UNIX COMMANDS

**ALGORTIHM:**

STEP1 : Start the program

STEP2 : Open the directory with directory object dp STEP3 : Read the directory content and print it.

STEP4: Close the directory.

# PROGRAM:

#include<stdio.h> #include<dirent.h> main(int argc, char \*\*argv)

{

DIR \*dp;

struct dirent \*link; dp=opendir(argv[1]);

printf(“\n contents of the directory %s are \n”, argv[1]); while((link=readdir(dp))!=0)

printf(“%s”,link->d\_name); closedir(dp);

}

# PROGRAM FOR SIMULATION OF GREP UNIX COMMANDS ALGORITHM

STEP1: Start the program

STEP2: Declare the variables fline[max], count=0, occurrences=0 and pointers \*fp,

\*newline**.**

STEP 3: Open the file in read mode.

STEP4: In while loop check fgets(fline,max,fp)!=NUL

STEP 5: Increment count value.

STEP 6: Check newline=strchr(fline, „\n‟)

STEP 7: print the count,fline value and increment the occurrence value. STEP 8: Stop the program

# PROGRAM:

#include<stdio.h> #include<string.h> #define max 1024 void usage()

{

printf(“usage:\t. /a.out filename word \n “);

}

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

{

FILE \*fp;

char fline[max]; char \*newline; int count=0;

int occurrences=0; if(argc!=3)

{

usage();

exit(1);

}

if(!(fp=fopen(argv[1],”r”)))

{

printf(“grep: couldnot open file : %s \n”,argv[1]); exit(1);

}

while(fgets(fline,max,fp)!=NULL)

{

count++; if(newline=strchr(fline, „\n‟))

\*newline=‟\0‟; if(strstr(fline,argv[2])!=NULL)

{

printf(“%s: %d %s \n”, argv[1],count, fline);

DEPARTMENT OF CSE, GGV, LAB MANUAL OF OPERATING SYSTEM FOR B.TECH. Page 13

occurrences++;

}

}

}

DEPARTMENT OF CSE, GGV, LAB MANUAL OF OPERATING SYSTEM FOR B.TECH. Page 14