Disk utilities:

Df,du,

Process utilities:

Ps,top,nice,rence,kill,

N/W:ifconfig,hostname,netstat,traceroute,finger,telnet.

Ex.No:2 Programs using the following system calls of UNIX operating system fork, exec, getpid, exit, wait, close, stat, opendir, readdir

1. PROGRAM FOR SYSTEM CALLS OF UNIX OPERATING SYSTEMS (OPENDIR, READDIR, CLOSEDIR

ALGORITHM: STEP 1: Start the program.

STEP 2: Create struct dirent.

STEP 3: declare the variable buff and pointer dptr.

STEP 4: Get the directory name.

STEP 5: Open the directory.

STEP 6: Read the contents in directory and print it.

STEP 7: Close the directory.

PROGRAM:

// Unix system calls

// opendir, readdir

#include <stdio.h>

#include <dirent.h>

int main()

{

struct dirent \*de;

DIR \*dr = opendir(".");

if (dr == NULL)

{

printf("Could not open current directory" );

return 0;

}

while ((de = readdir(dr)) != NULL)

printf("%s\n", de->d\_name);

closedir(dr);

return 1;

}

o/p:

2. PROGRAM FOR SYSTEM CALLS OF UNIX OPERATING SYSTEM (fork, getpid, exit)

ALGORITHM:

STEP 1: Start the program.

STEP 2: Declare the variables pid,pid1,pid2.

STEP 3: Call fork() system call to create process.

STEP 4: If pid==-1, exit.

STEP 5: Ifpid!=-1 , get the process id using getpid().

STEP 6: Print the process id.

STEP 7:Stop the program

Spce for fork,------etc system calls 1 or 2pages or write end last program.

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

Cp

1.Program for simulation of cp unix commands

create Directory

Ex $mkdir first

$cd first

$pwd

/root/first

$gedit cp.c

1.Write the PROGRAM source code

2,compile gcc cp.c

$touch tx1.txt tx2.txt

Cat >>tx1.txt

Os or unix lab

123

Abc

Ctrl+z

./a.out tx1.txt tx2.txt

$cat f1.txt enter display copy of file f1.txt file fields

Os or unix lab

123

Abc

**ALGORITHM:**

STEP1: Start the program

STEP 2:Declare the variables int main(int argc,char\* argv[])

STEP3: Open the file in read & write 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<stdio.h>

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

{

FILE \*fp1,\*fp2;

fp1=fopen(argv[1],”r”);

fp2=fopen(argv[2],”w”);

If(!fp1||!fp2||argc!=3)

{

printf(“ error”);

}

char c;

while((c=fgetc(fp1))!=EOF)

{

fputc(c,fp2);

}

fclose(fp1);

fclose(fp2);

}

o/p:

gcc cp.c

./a.out tx1.txt tx2.tx2

Result:successfully Copy the file source to destination.

Ls program:

2.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>

#include<stdlib.h>

int 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);

return 0;

}

o/p: gcc ls.c (mkdir venu,cd venu,mkdir os cd..)

./a.out

Enter Directory name

Venu

Sub folders also displayed

3. 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)!=NULL

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 or source code**

#include<stdio.h>

#include<dirent.h>

int main() {

char fn[10], pat[10], temp[200];

FILE \*fp;

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

scanf("%s", fn);

printf("Enter the pattern: ");

scanf("%s", pat);

fp = fopen(fn, "r");

while (!feof(fp)) {

fgets(temp, sizeof(fp), fp);

if (strcmp(temp, pat))

printf("%s", temp);

}

fclose(fp);

return 1;

}

O/p:

$gcc grep.c –o grep

$./a.out or grep

Enter File name:grep.c

Enter the pattern:while

Display complete program (while loop)