#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<fcntl.h>

#include<sys/stat.h>

#define MAX\_BUF 1024

int main()

{

int fd,c=0;

char \*fifo1="fifo1";

char \*fifo2="fifo2";

int fd1;

int words=1,lines=1,chars=0;

char buf1[MAX\_BUF];

mkfifo(fifo1,0666);

fd=open(fifo1,O\_RDWR);

char str;

printf("\nEnter the String:");

while ((str=getchar())!='#')

buf1[c++]=str;

buf1[c]='\0';

write(fd,buf1,sizeof(buf1));

close(fd);

unlink(fifo1);

fd1=open(fifo2,O\_RDWR);

read(fd1,buf1,sizeof(buf1));

printf("\nThe contents of file are %s\n",buf1);

int i=0;

while(buf1[i]!='\0')

{

if(buf1[i]==' '||buf1[i]=='\n')

{words++;}

else

{chars++;}

if(buf1[i]=='\n')

{ lines++;}

i++;

}

printf("\n No of Words: %d",words);

printf("\n No of Characters: %d",chars);

printf("\n No of Lines: %d",lines);

close(fd1);

return 0;

}

OUTPUT:

Enter the String:Hello my name is Isha Velankar

#

The contents of file are: Hello my name is Isha Velankar

No of Words: 7

No of Characters: 31

No of Lines: 2

2. SHARED MEMORY FOR READER PROCESS

INPUT

#include <sys/ipc.h>

#include <sys/shm.h>

#include <stdio.h>

int main()

{

// ftok to generate unique key

key\_t key = ftok("shmfile",65);

// shmget returns an identifier in shmid

int shmid = shmget(key,1024,0666|IPC\_CREAT);

// shmat to attach to shared memory

char \*str = (char\*) shmat(shmid,(void\*)0,0);

printf("Write Data : ");

gets(str);

printf("Data written in memory: %s\n",str);

//detach from shared memory

shmdt(str);

reader()

return 0;

}

void reader()

{

// ftok to generate unique key

key\_t key = ftok("shmfile",65);

// shmget returns an identifier in shmid

int shmid = shmget(key,1024,0666|IPC\_CREAT);

// shmat to attach to shared memory

char \*str = (char\*) shmat(shmid,(void\*)0,0);

printf("Data read from memory: %s\n",str);

//detach from shared memory

shmdt(str);

// destroy the shared memory

shmctl(shmid,IPC\_RMID,NULL);

}

Write Data : hello my name is John

Data written in memory: hello my name is John

Data read from memory: hello my name is John

OUTPUT:

Write Data : Hello my Name is isha

Data written in memory: Hello my Name is isha

Data read from memory: Hello my Name is isha