OS Lab 5

Utkarsh Bajaj

Ans 1)

producer.c

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<unistd.h>

#include<fcntl.h>

#include<limits.h>

#include<sys/types.h>

#include<sys/stat.h>

#define FILE\_NAME "/tmp/my\_fifo"

int main()

{

int pipe\_fd;

int res;

int open\_mode = O\_WRONLY;

int bytes\_sent = 0;

char buffer[4];

printf("Enter the Number\n");

scanf("%s",buffer);

if(access(FILE\_NAME,F\_OK)==-1)

{

res = mkfifo(FILE\_NAME,0777);

if(res!=0)

{

printf("Cannot create fifo\n");

exit(1);

}

}

printf("Process % d opening fifo\n", getpid());

pipe\_fd = open(FILE\_NAME,open\_mode);

printf("Process %d : Result %d\n",getpid(),pipe\_fd );

if(pipe\_fd!=-1)

{

res = write(pipe\_fd,buffer,4);

if(res==-1)

{

printf("Error writing on pipe\n");

exit(1);

}

(void)close(pipe\_fd);

}

else{

printf("Error openeing the pipe\n");

exit(1);

}

printf("Finished\n");

}

consumer.c

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<unistd.h>

#include<fcntl.h>

#include<limits.h>

#include<sys/types.h>

#include<sys/stat.h>

#define FILE\_NAME "/tmp/my\_fifo"

int main()

{

int pipe\_fd;

int res;

int open\_mode = O\_RDONLY;

int bytes\_sent = 0;

char buffer[4];

memset(buffer,'\0',sizeof(buffer));

printf("Process % d opening fifo\n", getpid());

pipe\_fd = open(FILE\_NAME,open\_mode);

printf("Process %d : Result %d\n",getpid(),pipe\_fd );

if(pipe\_fd!=-1)

{

res = read(pipe\_fd,buffer,4);

printf("Recieved : %s\n",buffer);

}

else{

printf("Error openeing the pipe\n");

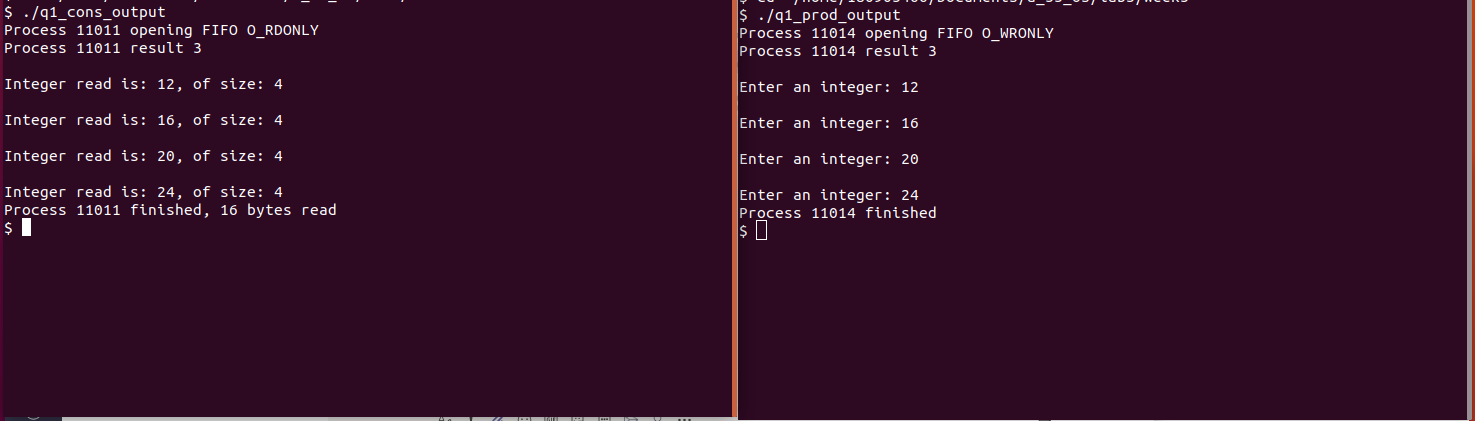
exit(1);

}

printf("Finished\n");

}

Sample i/o:



Ans 2)

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int main()

{

int n;

int fd[2];

char buf[201];

char \*data = "Hello World\n";

pipe(fd);

write(fd[1],data,strlen(data));

if((n = read(fd[0],buf,200))>=0)

{

buf[n] = 0;

printf("Read %d bytes from the pipe : %s\n",n,buf );

}

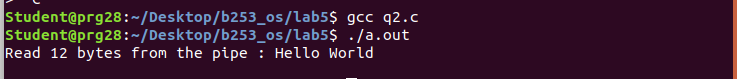
else{

perror("read");

exit(1);

}

}



ans 3)

read file

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<unistd.h>

#include<fcntl.h>

#include<limits.h>

#include<sys/types.h>

#include<sys/stat.h>

#define FILE\_NAME "/tmp/my\_fifo1"

int main()

{

int pipe\_fd1,pipe\_fd2;

int res;

char buffer1[100],buffer2[100];

while(1)

{

pipe\_fd1 = open(FILE\_NAME,O\_RDONLY);

res = read(pipe\_fd1,buffer1,sizeof(buffer1));

if(strcmp(buffer1,"exit")==0)

break;

if(res==-1)

{

printf("Error reading from pipe\n");

exit(1);

}

printf("Recieved from file 1 : %s\n",buffer1);

pipe\_fd2 = open(FILE\_NAME,O\_WRONLY);

printf("Enter string : \n");

gets(buffer2);

res = write(pipe\_fd2,buffer2,strlen(buffer2)+1);

if(strcmp(buffer2,"exit")==0)

break;

if(res==-1)

{

printf("Error writing on pipe\n");

exit(1);

}

}

}

write file

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<unistd.h>

#include<fcntl.h>

#include<limits.h>

#include<sys/types.h>

#include<sys/stat.h>

#define FILE\_NAME "/tmp/my\_fifo1"

int main()

{

int pipe\_fd1,pipe\_fd2;

int res;

char buffer1[100],buffer2[100];

if(access(FILE\_NAME,F\_OK)==-1)

{

res = mkfifo(FILE\_NAME,0777);

if(res!=0)

{

printf("Cannot create fifo\n");

exit(1);

}

}

while(1)

{

pipe\_fd1 = open(FILE\_NAME,O\_WRONLY);

printf("Enter string : \n");

gets(buffer1);

res = write(pipe\_fd1,buffer1,strlen(buffer1)+1);

if(strcmp(buffer1,"exit")==0)

break;

if(res==-1)

{

printf("Error writing on pipe\n");

exit(1);

}

pipe\_fd2 = open(FILE\_NAME,O\_RDONLY);

res = read(pipe\_fd2,buffer2,sizeof(buffer2));

if(strcmp(buffer2,"exit")==0)

break;

if(res==-1)

{

printf("Error reading from pipe\n");

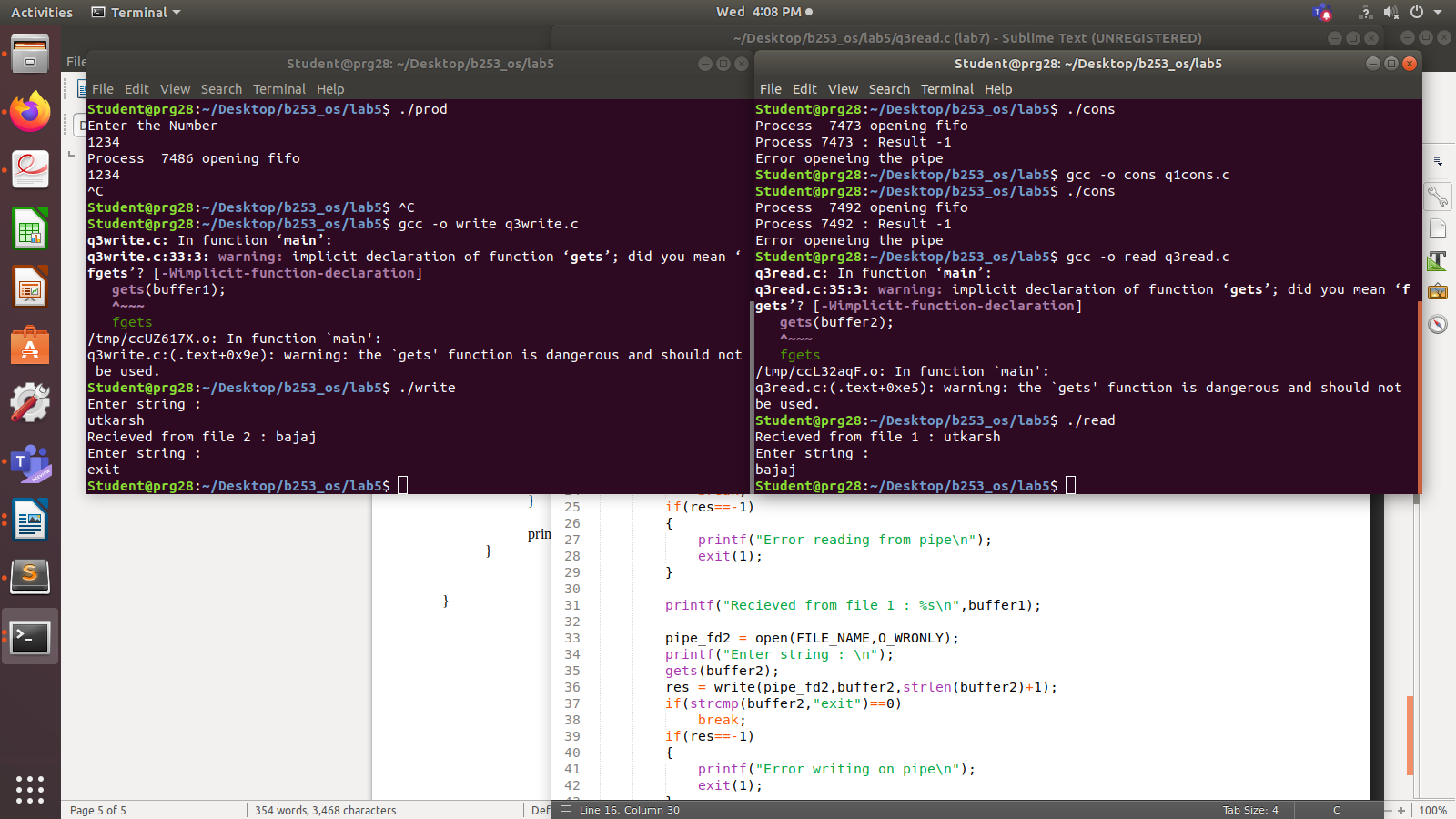
exit(1);

}

printf("Recieved from file 2 : %s\n",buffer2 );

}

}



ans 4)

write

#include <stdio.h>

#include <stdlib.h>

struct threeNum

{

int n1, n2, n3;

};

int main()

{

int n = 1;

struct threeNum num;

FILE \*fptr;

if ((fptr = fopen("rbi.bin","wb")) == NULL){

printf("Error! opening file");

exit(1);

}

num.n1 = n;

num.n2 = 5\*n;

num.n3 = 5\*n + 1;

fwrite(&num, sizeof(struct threeNum), 1, fptr);

fclose(fptr);

return 0;

}

read

#include <stdio.h>

#include <stdlib.h>

struct threeNum

{

int n1, n2, n3;

};

int main()

{

int n;

struct threeNum num;

FILE \*fptr;

if ((fptr = fopen("rbi.bin","rb")) == NULL){

printf("Error! opening file");

exit(1);

}

fread(&num, sizeof(struct threeNum), 1, fptr);

printf("n1: %d\tn2: %d\tn3: %d\n", num.n1, num.n2, num.n3);

fclose(fptr);

return 0;

}

