Name – Shubham Pawar

Roll No – Shubham Pawar

Subject – OS

Div – A

Q . Write a program for one way communication through pipe.

Ans –

#include <stdio.h>

#include <unistd.h>

int main(){

int pipefds[2];

int returnStatus;

// Reading and the message to be write

char readMessage[20];

//creating the pipe

returnStatus = pipe(pipefds);

//checking if the pipe is created

if(returnStatus !=0){

printf("Pipe was unable to create");

return 1;

}

/\*

pipefds[1] is used to write the message

pipefds[0] is used to write the message

\*/

/\*

########################################

Using the hardcoded values to the pipe

########################################

char writeMessage[2][20] = {"Hii ","Hello"};

// Writing the 1st message to he pipe

printf("Writing message 1 to the pipe : %s \n",writeMessage[0]);

write(pipefds[1],writeMessage[0],sizeof(writeMessage[0]));

read(pipefds[0],readMessage,sizeof(readMessage));

printf("Message Read from Pipe: %s",readMessage);

// writing the 2nd message to the pipe

printf("Writing message 2 to the pipe : %s \n",writeMessage[1]);

write(pipefds[1],writeMessage[1],sizeof(writeMessage[1]));

read(pipefds[0],readMessage,sizeof(readMessage));

printf("Message Read from Pipe: %s",readMessage);

\*/

/\*

##################################################

Using Values from the user

##################################################

\*/

//reading the values from the user

char message1[20];

char message2[20];

printf("Enter the message 1:");

scanf("%s",message1);

printf("Enter the message 2:");

scanf("%s",message2);

// writing the message to the pipe

printf("Message 1: %s\n",message1);

write(pipefds[1],message1,sizeof(message1));

read(pipefds[0],readMessage,sizeof(readMessage));

printf("Message Read : %s \n",readMessage);

//Writing and reading Message 2 to the Pipe

printf("Message 2: %s\n",message2);

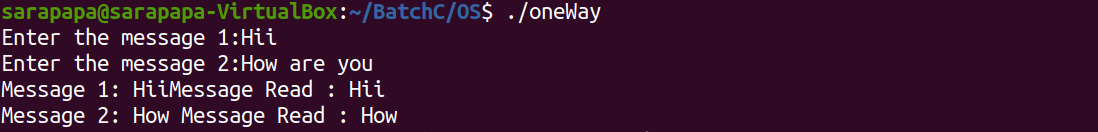
write(pipefds[1],message2,sizeof(message2));

read(pipefds[0],readMessage,sizeof(readMessage));

printf("Message Read : %s \n",readMessage);

return 0;

}



Q . Write a program for two way communication through pipe.

Ans –

#include<stdio.h>

#include<unistd.h>

int main() {

int pipefds1[2], pipefds2[2];

int returnstatus1, returnstatus2;

int pid;

char pipe1writemessage[20] = "Hi";

char pipe2writemessage[20] = "Hello";

char readmessage[20];

returnstatus1 = pipe(pipefds1);

if (returnstatus1 == -1) {

printf("Unable to create pipe 1 \n");

return 1;

}

returnstatus2 = pipe(pipefds2);

if (returnstatus2 == -1) {

printf("Unable to create pipe 2 \n");

return 1;

}

pid = fork();

if (pid != 0) { // Parent proess

close(pipefds1[0]); // Close the unwanted pipe1 read side

close(pipefds2[1]); // Close the unwanted pipe2 write side

/\*

Reading the Inout from the user

\*/

char message1[20];

printf("Enter the Writing message for pipe 1:");

scanf("%s",message1);

//printf("In Parent: Writing to pipe 1 – Message is %s\n", pipe1writemessage);

write(pipefds1[1], message1, sizeof(pipe1writemessaiiiiige));

read(pipefds2[0], readmessage, sizeof(readmessage));

printf("In Parent: Reading from pipe 2 – Message is %s\n", readmessage);

} else { //child process

close(pipefds1[1]); // Close the unwanted pipe1 write side

close(pipefds2[0]); // Close the unwanted pipe2 read side

read(pipefds1[0], readmessage, sizeof(readmessage));

printf("In Child: Reading from pipe 1 – Message is %s\n", readmessage);

/\*

Writing to pipe 2

\*/

char message2[20];

printf("Enter the message for pipe 2:");

scanf("%s",message2);

// printf("In Child: Writing to pipe 2 – Message is %s\n", pipe2writemessage);

write(pipefds2[1], message2, sizeof(pipe2writemessage));

}

return 0;

}

Text

Description automatically generated