T2|287
Jayesh Patil TE IT

Assignment no - 7(a)

Aim :- full duplex communiccation between two independent Process. first Process accepts senter and writes on one Pipe to be read by second Process and second Process counts number of lines in accepts sentences write this output in txt file and writes the contents of the file and write the contents of dhe file on second pipe to be read by first Process and displays on standard output.

objectives :- To study
- FIFO
- FIFO operation
- use of fifo for inter Process communi. cation

Theory :-

PIPES :-
Pipes Provides an inter-Process communicaation channel retween related Processes. The Pipe interdonce is intented to look a file interface although a pipe may seem like a file it is not a file. Each write to Pace pipe fills as man does ~~no~~ blacks as are needed to satisfy it Provideal dhat it does not exceed the maximum Pipe size filled black are because they do not exist any where in file system creating a pipe.

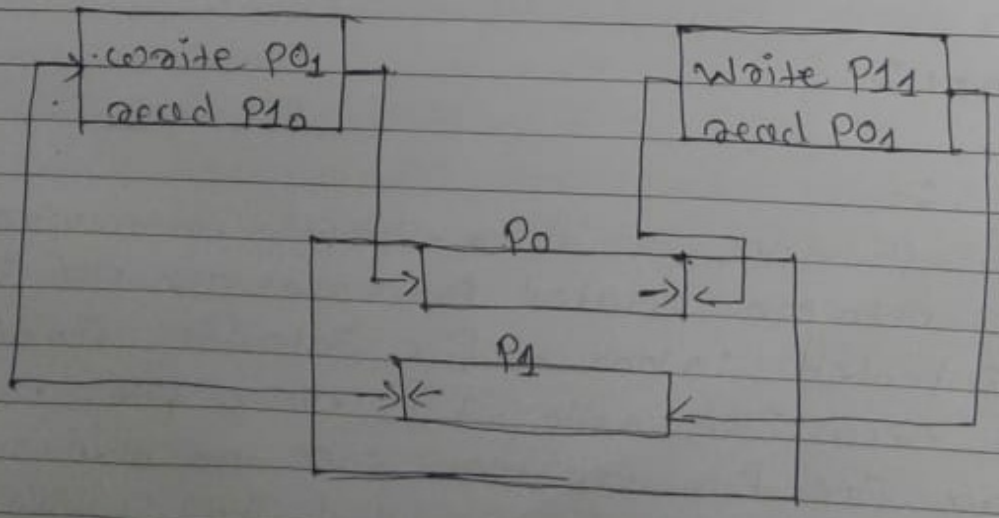
ssize_t read(int fd, void *buf, size_t count);

Read upto count bytes from file descriptor fd into buffer starting at buf

A UNIX pipe provides half duplex communication The pipe (2) system call return two file descriptor from a "pipe" a one-way communication channel with "read-end" and "write end"

read(fds[0] buf, len) write(fds[1] buf, len)

A unix pipe provide full duplex communication v: two pipes one for each direction create two sep pipes say P0 and P1



FIFOs

A FIFO (first in first out) in one way flow of data fifo have a name so unrelated process can share the fifo, fifo is named pipe Any process can open or close the fifo

Properties

1) After a fifo is created, it can be opened for read or write.
2) Normally, opening a fifo for read or write it blocks unit another process opens it for write or read
3) A read get as much as it requests or much data as fifo has. whichever is less.
4) A write to fifo is atomic, as along as write does not exceed the capacity of the fifo
5) fifo must be opened by two processes, one opens it as reader on one end, the other opens it as sender on other end the first / earlier opener has to wait until the second / later opener to come this is some what like hand-shaking.

Conclusion:-

Thus we studied Inter Process communication using fifos