

# OS 실습 6주차

201802061 김민환

## 실습1 PIPE

```
os_201802061@OS00:~$ cd week6
os_201802061@OS00:~/week6$ ls
fifo_temp input input.c output output.c recv
os_201802061@OS00:~/week6$ ./input
hello
os_201802061@OS00:~/week6$ ./output
finethxu
string : finethxu
os_201802061@OS00:~/week6$ ./input | ./output
string : hello
os_201802061@OS00:~/week6$ ./input > msg.txt
os_201802061@OS00:~/week6$ cat msg.txt
hello
os_201802061@OS00:~/week6$ ./output < msg.txt
string : hello
os_201802061@OS00:~/week6$
```

input : hello를 출력하는 프로세스

output : 입력받은 문자열을 출력하는 프로세스

| ( pipe ) : 프로세스의 출력을 다른 프로세스의 입력으로 전달한다

> , < : 리디렉션 , 입출력 방향을 설정한다

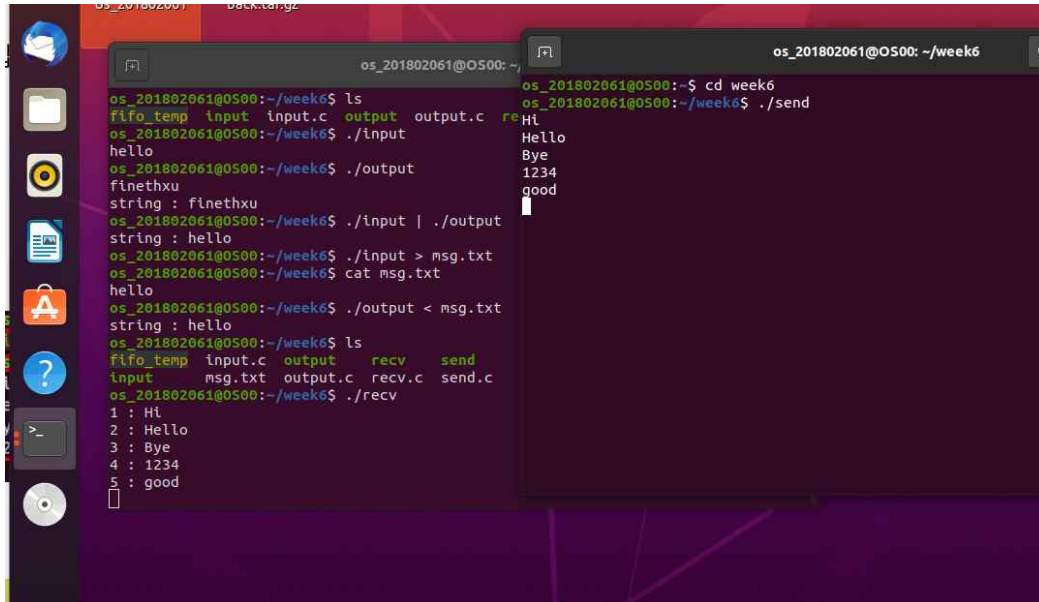
input의 출력 값 hello가 output의 입력값으로 되어 hello가 출력된다.

msg.txt에 input의 출력값이 작성된다.

output에 msg.txt의 텍스트가 버퍼에 입력되어 출력된다.

각각 pipe 로 프로세스 간 단방향 통신을 하였다

## 실습2 FIFO



```
os_201802061@OS00: ~/week6
os_201802061@OS00: ~/week6$ ls
fifo_temp  input  input.c  output  output.c
os_201802061@OS00: ~/week6$ ./input
hello
os_201802061@OS00: ~/week6$ ./output
finethxu
string : finethxu
os_201802061@OS00: ~/week6$ ./input | ./output
string : hello
os_201802061@OS00: ~/week6$ ./input > msg.txt
os_201802061@OS00: ~/week6$ cat msg.txt
hello
os_201802061@OS00: ~/week6$ ./output < msg.txt
string : hello
os_201802061@OS00: ~/week6$ ls
fifo_temp  input.c  output  recv  send
input      msg.txt  output.c  recv.c  send.c
os_201802061@OS00: ~/week6$ ./recv
1 : Hi
2 : Hello
3 : Bye
4 : 1234
5 : good
```

send : FIFO\_FILE을 WriteOnly로 open 해서 fgets()로 받은 입력 값을 파일에 작성하는 프로세스

recv : FIFO\_FILE을 mkfifo()로 생성한 뒤 반복해서 ReadOnly로 open해서 읽어들이는 프로세스

send로 스트링을 입력하면 fifo\_temp 버퍼에 저장되고 recv가 write하며 버퍼를 비운다  
커널 버퍼를 임시 저장 공간으로 사용하기 때문에 양방향 통신이 가능하게된다