상명대학교 컴퓨터과학과 "HAEA0014: 유닉스 프로그래밍" 실습 지침서

- 실습 번호: lab-07

- 실습 디렉토리: ~/unix/lab-07 이름: \_\_\_\_\_\_\_\_\_\_\_정경재\_\_\_\_\_\_\_\_\_

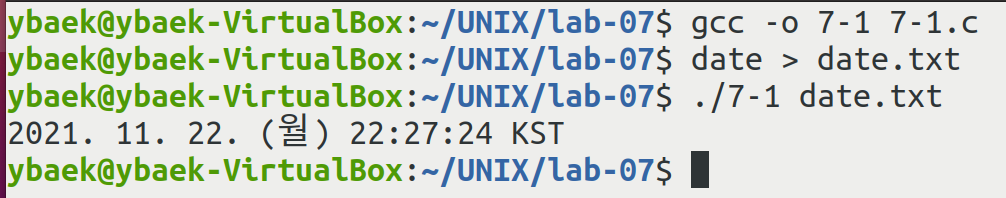
- 실습 날짜: 2021년 11월 28일

- 실습 제목: mmap, pipe

- 실습 내용: 학번: \_\_\_\_\_\_\_\_201810993\_\_\_\_\_\_\_\_\_

(모든 프로그램에는 각자의 comment를 추가하시오. 과정과 결과를 캡처하여 넣으시오.)

1. 8장 slide의 8-1.c를 참고하여 command-line argument로 주어진 파일을 mmap()를 이용하여 메모리에 매핑하고 이를 출력하는 프로그램을 7-1.c로 만들고 이를 컴파일하여 test하되 아래와 같이 하시오.



7-1.c를 제출하시오.

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

2. 8장 slide의 8-5.c를 참조하고, 위 7-1.c를 수정하여

1) parent process가 메모리 맵을 만들어 argv[1] 에 해당하는 파일을 매핑하고

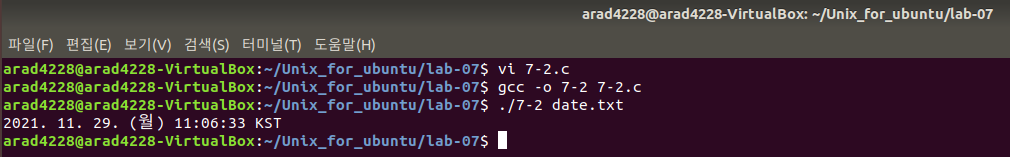
2) child process가 그 메모리 맵의 내용을 stdout으로 출력하는

프로그램을 7-2.c로 만들고 이를 컴파일하고 test하여 결과를 확인하시오.

7-2.c를 제출하시오.

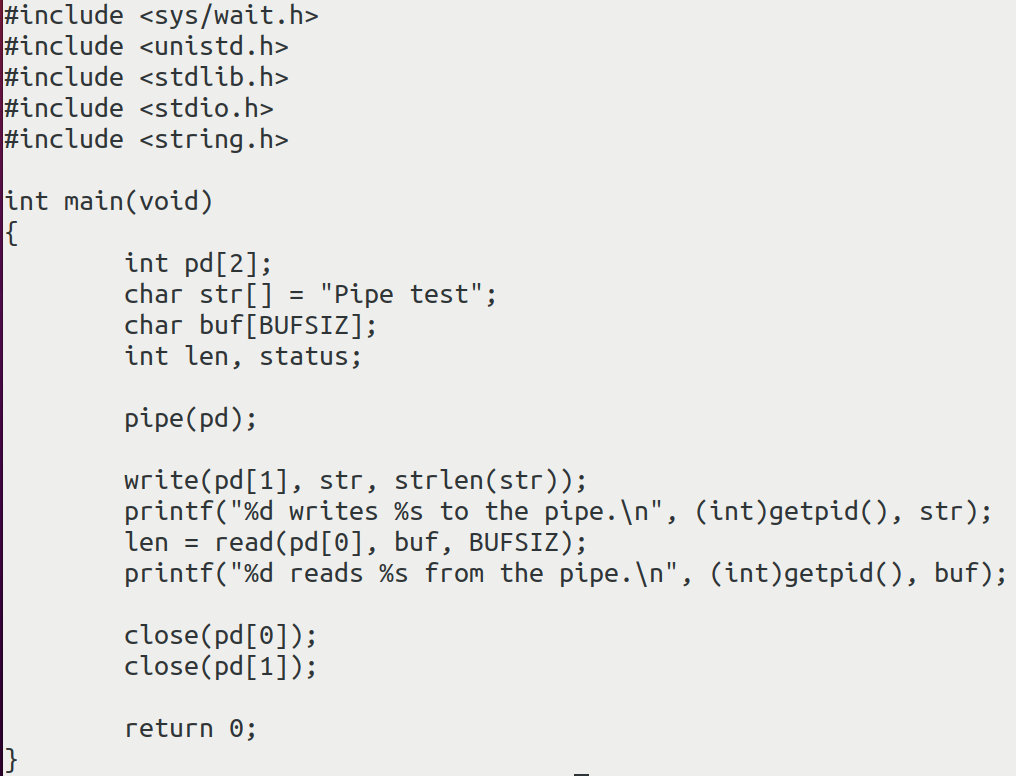
텍스트이(가) 표시된 사진

자동 생성된 설명



3.. pipe()를 테스트하는 아래의 프로그램을 헤더 파일을 찾아 넣고, 에러 처리도 하도록 수정 하여 7-3.c로 저장하시오. 컴파일하여 실행하고 왜 그런 결과가 나오는지를 설명하시오.

7-3.c를 제출하시오.



텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

6326번 프로세스 즉 7-3을 실행하는 프로세스가 pipe를 통해pd[0](read),pd[1](write)를 생성하고 해당 pipe의 read인 pd[1]에 write를 통해 원하던 문자열을 집어넣고 read를 통해 pd[0]에서 문자열을 읽었으므로 해당 결과가 나오게 된다.

4. 위의 7-3.c를 fork()를 이용, 수정하여 Parent Process에서 pipe로 “Pipe Test”를 보내고 Child Process에서 pipe를 읽어 그 내용을 출력하는 프로그램 7-4.c를 만들고 컴파일하여 실행하시오.

7-4.c를 제출하시오.

텍스트이(가) 표시된 사진

자동 생성된 설명

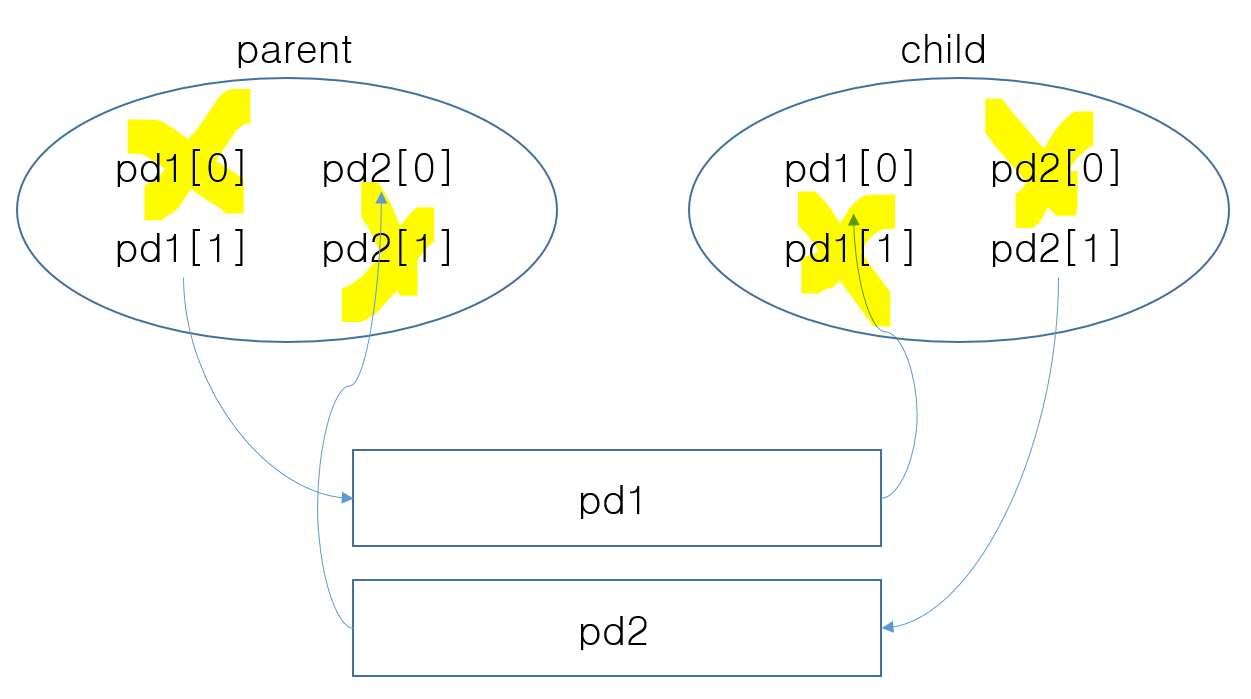
텍스트, 스크린샷, 화면이(가) 표시된 사진

자동 생성된 설명

5. 두 개의 파이프를 pipe(pd1), pipe(pd2)를 통해 만들고

Parent process는 pd1[1]에 write하고 pd2[0]를 읽도록,

Child process는 pd2[1]에 write하고 pd1[0]를 읽도록 아래 그림과 같이 구성하시오.



parent는 child에게 “from parent”라는 문자열을 write하고 child는 parent에게 “from child”라는 문자열을 write한 후 각각 받은 문자열을 stdout으로 출력하는 프로그램 7-5.c를 작성하고 수행하여 결과를 보이시오.

7-5.c를 제출하시오.

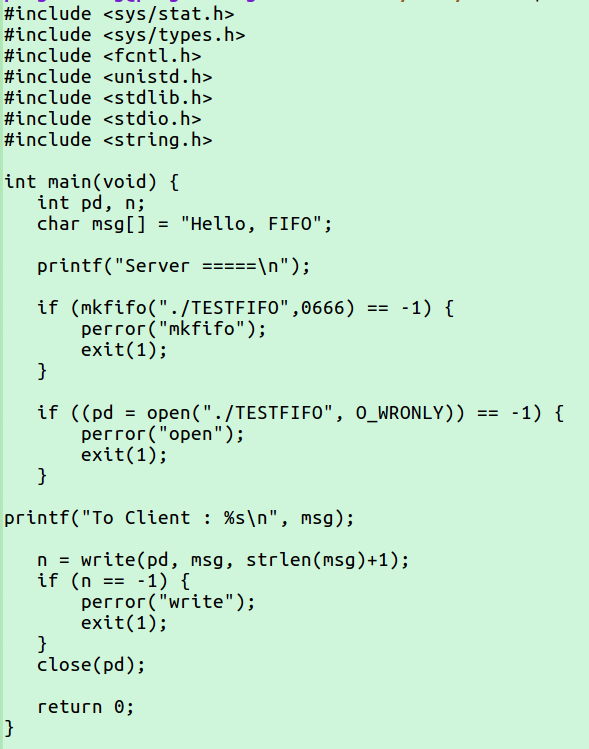
텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

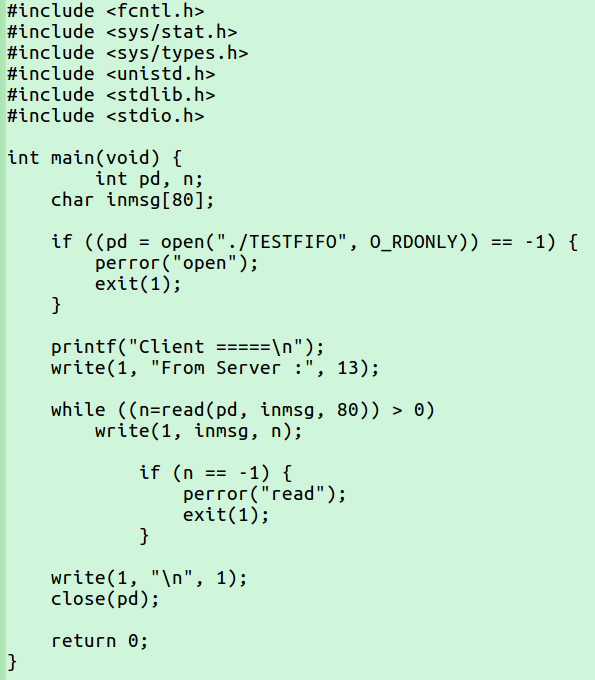
6. 서버를 담당하는 7-6-s.c는 아래와 같이 작성한다.



텍스트이(가) 표시된 사진

자동 생성된 설명

클라이언트를 담당하는 7-6-c.c는 아래와 같이 작성한다.



텍스트이(가) 표시된 사진

자동 생성된 설명

7-6-s.c는 7-6-s로 7-6-c.c는 7-6-c로 컴파일한 후, 두 프로그램을 실행시켜 결과를 보이고 설명하시오.

텍스트이(가) 표시된 사진

자동 생성된 설명

7-6-s 에서 이름이 있는 파이프 즉 TESTFIFO를 사용해 파이프에 원하는 메시지를 작성하고 클라이언트에서는 이 TESTFIFO에 대한 내용을 읽어 출력했기에 해당 결과가 발생합니다.

끝.