

실습 3. 파일 처리, 프로세스 및 프로세스간 통신

1. 실습의 목표

본 실습에서는 리눅스에서 파일 처리, 프로세스 및 프로세스간 통신 기법에 대한 기본 개념을 확립하고 프로그래밍 및 설계 능력을 갖추는 것이 주요 목표이다.

2. 실습에 필요한 기초 지식

본 실습에서는 파일 및 디렉토리 관련 함수들, 리눅스 프로세스와 프로세스간 통신 관련 함수들에 대한 이해와 지식을 필요로 한다.

3. 실습에 필요한 장비

하드웨어

1	PC	Host Computer
---	----	---------------

소프트웨어

1	리눅스	사용 환경
1	gcc	GNU 컴파일러 패키지
1	make	프로그래밍 도구
1	git	버전 관리 시스템
1	gdb	GNU 디버거

4. 실습에 필요한 준비 사항

1. 파일 처리 함수에 대하여 조사하고 이해한다.
2. 입출력 장치 파일에 대하여 조사하고 요약하여 보고서에 정리하여 본다.
3. 교재에서 프로세스 및 시그널에 대한 내용을 읽고 이해한다.
4. 리눅스의 프로세스 모델과 시그널/쓰레드와의 관계에 대하여 조사하고 요약하여 보고서에 정리하여 본다.
5. 교재에서 프로세스 간 통신에 대한 내용을 읽고 이해한다.
6. 리눅스에서 비동기 통신 기법에 대하여 조사하고 요약하여 보고서에 정리하여 본다.

5. 실습 순서

1. 자신의 github 저장소에 lab3 프로젝트를 생성하고 아래의 모든 과제 프로그램을 업로드한다.
2. 파일 및 디렉토리와 관련된 함수들을 사용하여 프로그램을 작성하고 실행하여 보고, 익숙해지도록 사용해 본다.
3. 주어진 디렉토리 내에 존재하는 파일과 디렉토리를 나열하고, 디렉토리의 경우 재귀적으로 방문해서 그 디렉토리 내에 존재하는 파일과 디렉토리를 나열하는 프로그램을 작성하시오. 즉, "ls -R" 명령과 동일한 결과를 보이도록 하시오.
4. 몇 개의 문장을 타자하도록 하여 잘못 타이핑한 횟수와 평균 분당 타자수를 측정하는 타자 연습 프로그램을 구현하여 보시오.
5. 프로세스와 관련된 함수들을 사용하여 프로그램을 작성하고 실행하여 보고, 익숙해지도록 사용해 본다.
6. system 함수는 셸 명령이 실행되도록 하는데, 예를 들면, system("ls -la") 을 호출하면, 현재 디렉토리의 파일들을 나열해 준다. 이와 같은 기능을 수행하는 함수를 직접 구현하여 보자. 또, 이 함수를 이용하는 예제 프로그램을 통해서 "a.out ls -la" 와 같이 명령이 잘 동작하도록 해 보자.
7. 시그널과 관련된 함수들을 사용하여 프로그램을 작성하고 실행하여 보고, 익숙해지도록 사용해 본다.
8. 프로세스 간 통신 함수들을 사용하여 프로그램을 작성하고 실행하여 보고, 익숙해지도록 사용해 본다.
9. 메시지 큐를 사용하여 텍스트 기반의 간단한 채팅 프로그램을 구현하시오.
10. 공유 메모리를 사용하여 한 파일을 다른 파일로 복사하는 프로그램을 작성하시오. 단, 부모(읽는 프로세스)와 자식(쓰는 프로세스)프로세스가 공유 메모리 영역을 동시에 접근하는 일이 없도록 세마포어 같은 동기화 기법을 활용하시오.
11. (팀과제) 간단한 셸 프로그램을 만들고 다음과 같이 동작하도록 수정하시오.
 1. "exit"를 치면 프로그램을 끝내도록 프로그램을 수정하시오.
 2. csh, bash 등에서처럼 셸 명령의 마지막에 '&'을 입력하면 백그라운드로 실행되도록 프로그램을 수정하시오.
 3. csh, bash 등에서처럼 인터럽트키 (SIGINT: Ctrl-C, SIGQUIT: Ctrl-Z)가 동작하도록 프로그램을 수정하시오.
 4. 파일 재지향 (>, <) 및 파이프(|) 기능이 가능하도록 프로그램을 수정하시오.
 5. ls, pwd, cd, mkdir, rmdir, ln, cp, rm, mv, cat 명령을 팀원이 공평하게 나누어 구현하시오.

6. 검토

실습의 요점, 결과, 검토 사항에 대해 기술하라.