

Unix 프로그래밍 및 실습

기말과제

기말과제 #1

- ▶ 2장 ~ 9장까지 나온 명령, system call, 함수 요약 정리
 - ▶ 이번 학기에서 배운 내용을 총정리하는 의미
 - ▶ 각 장의 2절부터 순서대로 본인이 **요약**하고, 정리 (한글, 워드 등)
 - ▶ 제출기한 : 기말고사 당일 메일로 제출
 - ▶ 제목 : Unix2 기말 - 총정리

기말과제 #2 (1)

▶ 카운터에서 주문 받기

▶ 카운터에서는 주문 입력 기능을 수행

1. 카운터 번호 입력
2. 화면에 메뉴 출력 (메뉴 번호, 이름, 가격) – 본인이 원하는 자유 메뉴
3. 사용자에게 원하는 메뉴 번호, 수량 입력
4. 계속 주문 여부 출력
5. 계속 주문인 경우 2부터 반복
6. 더 이상 주문이 없는 경우 주문표(카운터 번호, {메뉴 번호, 수량}+, 총 음식 수) 출력
7. 주문표 파일(20141130.order)에 현재 시간(시, 분, 초)과 출력 내용 저장
 - ▶ 이때 주문표 파일 이름은 현재 시간을 읽어와서 그중 연도, 월, 일로 생성 (프로그램 상에서 생성)
 - ▶ 기존에 해당 파일이 있는 경우 append 모드로 저장

```
12:05:13 3번 테이블 2번 3개 4번 2개 총 5개
12:06:02 5번 테이블 7번 음식 1개 총 1개
...
```

기말과제 #2 (2)

- ▶ 각 단계별로 화면 캡처
- ▶ 프로그램 소스(화면 캡처가 아님!)와 주문표 파일들 같이 제출
 - ▶ 프로그램 개발 과정에서 만들어진 파일 모두 송부
- ▶ 프로그램 내 본인이 작성한 함수별로 설명 반드시 포함

기말과제 #3 (1)

▶ 카운터 - 주방

- ▶ 카운터 프로세스는 #2를 확장하여 주방 프로세스로 주문 전달
- ▶ 사용자로부터 주문을 받거나, 요리가 나오면 전달하는 메시지 출력
 - ▶ 메시지에는 요리 번호, 이름, 수량, 총 금액이 포함되어야 함
- ▶ 주방 프로세스
 - ▶ 전달된 주문 시간과 내용 파일로 저장
 - ▶ 파일이름은 20141130.cook
 - ▶ 조리 시작 (alarm과 linked list 이용)
 - ▶ 주문된 음식 중 제일 시간이 많이 걸리는 음식의 시간으로 alarm 설정
 - ▶ 음식이 준비되면 현재 시간과 해당 내용 파일에 저장하고 카운터 프로세스에게 내용 전달

```
12:05:13 3번 테이블 2번 음식 3개 주문
12:06:02 5번 테이블 7번 음식 1개 주문
12:0607 5번 테이블 2번 음식 1개 주문
12:10:14 3번 테이블 2번 음식 3개 조리완료
...
```

기말과제 #3 (2)

- ▶ 일단 음식 조리가 시작되면 모든 음식의 조리가 끝날 때까지 SIGINT는 불가
- ▶ 주문이 없는 상태에서 SIGINT가 들어오면 후속 조치(로그 파일, 파이프, FIFO 등 close) 후 종료
- ▶ 각 단계별 화면 캡처
- ▶ 프로그램 소스(화면 캡처가 아님!)와 요리 파일도 같이 제출
 - ▶ 프로그램 개발 과정 중 만들어진 요리 파일 모두 송부
- ▶ 카운터 프로세스와 주방 프로세스 설계 내용 설명 (특히 두 프로세스간 통신 방식에 대해 자세히 설명)
- ▶ 프로그램 내 본인이 작성한 함수별로 설명 반드시 포함

기말과제 #4 (1)

▶ 회계 프로그램

- ▶ #3 카운터 프로그램 확장하여 음식이 완료될 때마다 카운터번호, 요리완료 시간, 음식 번호, 수량을 파일에 저장 (file 이름 20141130.hst)
 - ▶ 기존에 해당 파일이 있는 경우 append 모드로 저장
- ▶ 카운터 프로그램에서 저장한 파일을 이용하여 일일 총 매출, 음식(번호)별 총 수량 및 판매금액, 카운터(번호)별 판매금액 계산하여 출력하는 프로그램 작성

기말과제 #4 (2)

- ▶ 카운터 프로그램에서 파일에 저장할 때 사용한 자료구조 설명
- ▶ 프로그램 소스(화면 캡처가 아님!)와 hst 파일 같이 제출
 - ▶ 프로그램 개발 과정 중 만들어진 모든 hst 파일 송부
- ▶ 프로그램 내 본인이 작성한 함수별로 설명 반드시 포함

기말 시험

- ▶ 시험 당일 응용과제별 채점표 배포 및 작성
- ▶ 주요 과제에 대한 질의, 응답, 시연
- ▶ 프로그램 작성은 본인이다!
 - ▶ 함수 사용이나 방식에 대한 토론은 가능하나, 코드는 자신이 직접 작성할 것
 - ▶ 부정행위 적발시 엄벌