

# 7장. 시그널

Unix 프로그래밍 및 실습

# 강의 내용

- 1절 개요
- 2절 시그널의 개념
- 3절 시그널 보내기
- 4절 시그널 기본 처리
- 5절 시그널 집합
- 6절 sigaction 함수의 활용
- 7절 알람 시그널
- 8절 기타 시그널 처리 함수
- <http://lily.mmu.ac.kr/lecture/13u2/ch07.pdf>
- 책에 나온 내용 반드시 man으로 확인할 것!
  - UNIX, LINUX 등 시스템마다 차이가 있을 수 있음을 반드시 인식

# 기본 실습 (1)

- [예제 7-1] ~ [예제 7-6] (각 10점)
  - 과제 개요 (2줄 이상)
  - 프로그램
  - 실행화면 캡처
- 보고서 첫 장에 다음 표 작성

과제	완성 여부 (O, X)	비고
예제 7-1 (10)	O	
예제 7-2 (10)	O	
...	O	
예제 7-6 (10)	X	

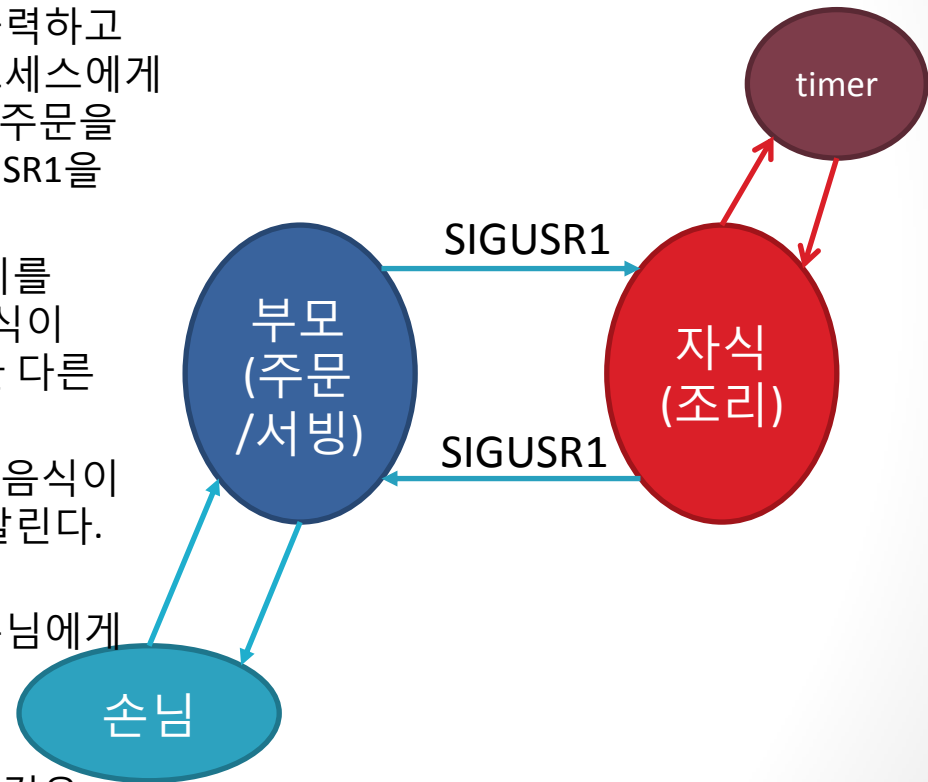
# 기본 실습 (2)

- [예제 7-7] ~ [예제 7-13] (각 10점)
  - 과제 개요 (2줄 이상)
  - 프로그램
  - 실행화면 캡처
- 보고서 첫 장에 다음 표 작성

과제	완성 여부 (O, X)	비고
예제 7-7 (10)	O	
예제 7-8 (10)	O	
...	O	
예제 7-13 (10)	X	

# 응용과제 #1 (1)

- 배점 : 1000점
- 부모 프로세스는 반복해서 메뉴를 출력하고 사용자로부터 주문을 받아 자식 프로세스에게 주문 내용을 알린다. (SIGUSR1) (일단 주문을 받으면 음식이 완료되기 전까지 SIGUSR1을 제외한 다른 시그널은 모두 무시)
- 자식 프로세스는 주문을 받으면 조리를 시작한다. (일단 조리를 시작하면 음식이 완성되기 전까지 SIGALARM을 제외한 다른 시그널은 모두 무시)
- 일정 조리시간이 지내면(alarm 이용) 음식이 완성되고, 이를 부모 프로세스에게 알린다. (SIGUSR1)
- 부모 프로세스는 음식이 완성되면 손님에게 알린다. (메시지 출력)
- 주문이 없는 상태에서 SIGINT를 받는 경우 메시지 출력하고 자식을 먼저 종료시키고 부모 종료



# 응용과제 #1 (2)

- 다음 확인표 작성

확인사항	완성 여부 (O, X)	비고
1. (부모) 자식 프로세스 생성 및 exec 정상 작동 여부		
2. (부모) 메뉴 출력 및 사용자로부터 입력 받기		
3. (부모) 주문 받기 전 Ctrl-C 받았을 때 정상 동작(메시지 출력 / 자식 프로세스 종료 / 자신 종료) 여부		
4. (부모) 주문 받은 후 관련 시그널 처리 준비한 후 자식에게 SIGUSR1 보내기		
5. (부모) 조리가 시작된 후 다른 시그널 무시		
6. (자식) 시그널 처리 준비		
7. (자식) SIGUSR1 받았을 때 관련 시그널 처리 준비한 후 조리 시작 (화면에 메시지 출력)		
8. (자식) SIGALARM 받았을 때 관련 시그널 처리 준비한 후 부모에게 통보		
9. (부모) 자식으로부터 SIGUSR1 받았을 때 관련 시그널 처리 준비한 후 손님에게 메시지 출력		
10. (부모) 음식이 완료된 후 Ctrl-C 받았을 때 정상 동작(메시지 출력 / 자식 프로세스 종료 / 자신 종료) 여부		

# 응용과제 #1 (3)

- 주문

- main
  - fork 후 부모이면
    - SIGINT 들어오면 자식도 종료하도록 핸들러 등록
    - while 루프
      - 화면에 주문 안내 출력
      - 주문이 들어오면 signal 처리함수들 등록
        - SIGUSR1 핸들러 등록
        - SIGINT 무시하도록 설정
      - SIGUSR1이 올 때까지 대기(pause)
    - fork 후 자식이면
      - exec로 프로그램 몸체 바꾸기
- SIGUSR1 핸들러
  - SIGUSR1이 들어오면 결과 출력
  - SIGINT 핸들러 설정
- SIGINT 핸들러
  - 자식에게 SIGINT 전달
  - wait 후 메시지 출력

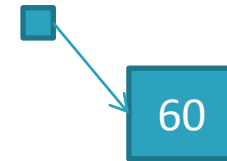
- 조리

- main
  - signal 처리함수들 등록
    - SIGUSR1 핸들러 등록
  - 루프 안에서 시그널 대기
- SIGUSR1 핸들러
  - SIGALRM 핸들러 등록
  - SIGINT 무시하도록 설정
  - 조리시간에 맞추어 alarm 호출
- SIGALRM 핸들러
  - SIGUSR1 보내기
  - SIGINT 허용하도록 설정

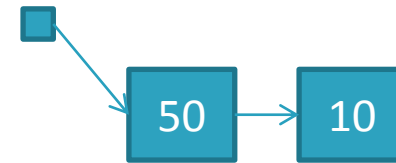
# 응용과제 #2

- 배점 : 1000점
- 응용 #1의 경우 자식 프로세스는 한번 조리가 시작되면 다음 주문을 못 받는 상황임.
- 조리가 시작된 후에도 주문을 계속 받아 조리를 할 수 있도록 개선하시오.
- 힌트
  - SIGALARM을 받기 전에 alarm(0) 을 하는 경우 남은 시간(초 단위)이 반환됨.
  - Linked list를 이용하여 각 요리의 남은 시간들을 관리

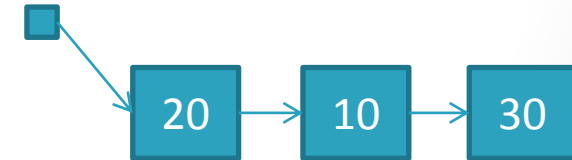
첫 주문



10초 후 2번째 주문



30초 후 3번째 주문



20초 후 SIGALARM 수신

