

시그널	기본처리	발생조건
SIGINT	종료	ctrl+c 입력시
SIGKILL	종료	강제 종료시
SIGQUIT	코어 덤프 (밑에 설명 있음)	ctrl + \입력시
SIGSTOP	중지 unblockable	????
SIGTSTP	중지	ctrl + z 입력시
SIGCONT	무시	중지된 프로세스 재개시

코어덤프(core dump)는 어떤 프로그램의 메모리 사용 상태를 특정 시점에서 기록한 것으로, 주로 파일로 만들어진다. 보통, 프로그램이 비정상적으로 종료되는 경우에 생기지만, 정상적으로 작동하는 프로그램으로부터도 코어덤프를 만들 수 있다. 예컨대, 기본설정을 변경하지 않은 유닉스에서는 SIGQUIT 시그널을 보냄으로써 코어덤프를 만들 수 있다.

signal(시그널종류, 시그널핸들러)

시그널 핸들러는 int를 인자로하고 void를 리턴하는 타입의 함수이다.

```

signal(SIGINT, SIG_IGN);
if((pid=fork())<0){
    perror("error fork()");
}
else if(pid==0){
    signal(SIGINT, SIG_DFL);
    while(1){
        printf("child =%d\n",i);
        i++;
        sleep(2);
    }
}

```

시그널 핸들러 자리에는 SIG_IGN, SIG_DFL가 들어갈 수 있다.

SIG_IGN은 해당 시그널을 무시하라는 것이고, SIG_DFL은 기본처리하라는 것이다.

위 예제를 보면, 포크이전에 SIGINT을 무시하도록 하고 자식만 기본처리로 바꾼다.

참고로 시그널 종류에 오는 건 _가 없고, 시그널핸들러에 오는 건 _가 있다.

필요헤더 : #include <signal.h>

kill(프로세스 id, 시그널종류)

필요헤더

#include<signal.h>