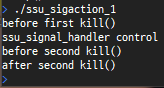
컴퓨터학부 20162448 김병준

1. 결과



1. 소스코드

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <unistd.h>  #include <signal.h>  void ssu\_signal\_handler(int signo) {  printf("ssu\_signal\_handler control\n");  }  int main(void) {  struct sigaction sig\_act;  sigset\_t sig\_set;  sigemptyset(&sig\_act.sa\_mask); // sigaction의 sa\_mask 시그널 집합 초기화  sig\_act.sa\_flags = 0;  sig\_act.sa\_handler = ssu\_signal\_handler; // sa\_handler에 핸들러 등록  sigaction(SIGUSR1, &sig\_act, NULL); // sig\_act에 SIGUSR1 시그널 등록  printf("before first kill()\n");  kill(getpid(), SIGUSR1); // 현재 프로세스에 전달  sigemptyset(&sig\_set); // sig\_set 시그널 집합 비움  sigaddset(&sig\_set, SIGUSR1); // sig\_set에 SIGUSR1 추가  sigprocmask(SIG\_SETMASK, &sig\_set, NULL); // 블록 시그널 등록  printf("before second kill()\n");  kill(getpid(), SIGUSR1); // 현재 프로세스에 전달  printf("after second kill()\n");  exit(0);  } |