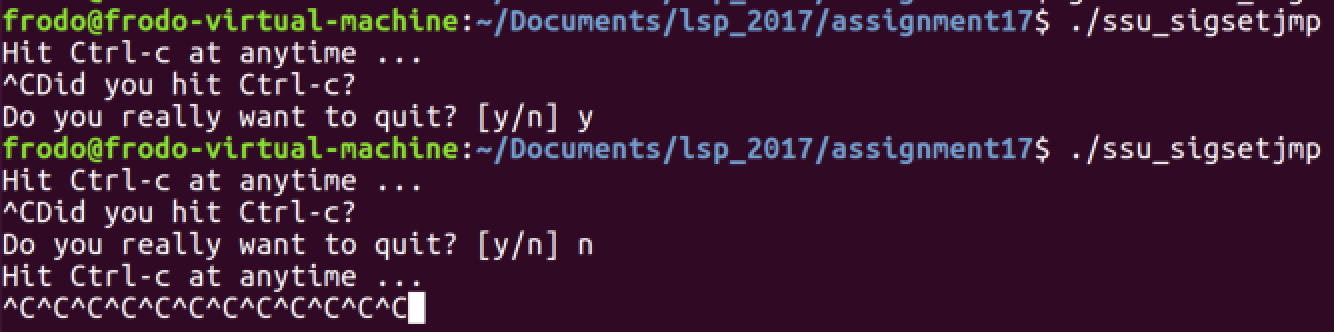
1. 소스코드

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <unistd.h>  #include <signal.h>  #include <setjmp.h>  void ssu\_signal\_handler(int signo);  jmp\_buf jump\_buffer;  int main(void) {  //SIGINT 시그널에 대한 시그널 핸들러 함수 등록  signal(SIGINT, ssu\_signal\_handler);  while(1) {  if(setjmp(jump\_buffer) == 0) {  //setjmp()가 직접 호출된 경우  printf("Hit Ctrl-c at anytime ... \n");  pause();  }  }  exit(0);  }  void ssu\_signal\_handler(int signo) {  char character;  //signo에 저장된 시그널을 무시  signal(signo, SIG\_IGN);  printf("Did you hit Ctrl-c?\n""Do you really want to quit? [y/n] ");  character = getchar();  if(character == 'y' || character == 'Y')  exit(0);  else {  //SIGINT 시그널에 대한 시그널 핸들러 함수 등록  signal(SIGINT, ssu\_signal\_handler);  //setjmp()로 이동  longjmp(jump\_buffer, 1);  }  } |

2. 실행결과



n을 입력한 경우 SIGINT를 발생시켜도 SIGINT가 블록된 상태이므로 시그널 핸들러 함수가 수행되지 않는다.