

Laboratório #3 - Sinais

1. Reproduza os experimentos descritos nos vídeo intitulado "**Advanced Programming in the UNIX Environment: Week 07, Segment 4 - Signals**" (URL - <https://youtu.be/Vh7rBGj0Ty4>), que são mostrados no intervalo de 10'07" até 16'05". Assistir a partir do instante 8'34" para reforçar alguns conceitos apresentados na sala de aula.

2. Gravar um vídeo mostrando a sua solução para a tarefa abaixo.

Descrição do tarefa: Implemente um programa em C/Linux que conta o número de CTRL-C que foi digitado e, na quinta vez, pergunta ao usuário se ele deseja realmente sair. Além disso, se o usuário digitar CTRL-Z, o número de CTRL-C deve ser exibido na tela.

3. Compile, teste e explique o funcionamento do programa abaixo.

```
#define _POSIX_SOURCE
#include <stdio.h>
#include <signal.h>
#include <time.h>
#include <unistd.h>

void catcher(int signum) {
    switch (signum) {
        case SIGUSR1: puts("catcher caught SIGUSR1");
                     break;
        case SIGUSR2: puts("catcher caught SIGUSR2");
                     break;
        default:      printf("catcher caught unexpected signal %d\n",
                             signum);
    }
}

main() {
    sigset_t sigset;
    struct sigaction sact;
    time_t t;

    if (fork() == 0) {
        sleep(10);
        puts("child is sending SIGUSR2 signal - which should be blocked");
        kill(getppid(), SIGUSR2);
        sleep(5);
        puts("child is sending SIGUSR1 signal - which should be caught");
        kill(getppid(), SIGUSR1);
    }
}
```

```

    exit(0);
}

sigemptyset(&sact.sa_mask);
sact.sa_flags = 0;
sact.sa_handler = catcher;
if (sigaction(SIGUSR1, &sact, NULL) != 0)
    perror("1st sigaction() error");
else if (sigaction(SIGUSR2, &sact, NULL) != 0)
    perror("2nd sigaction() error");
else {
    sigfillset(&sigset);
    sigdelset(&sigset, SIGUSR1);
    time(&t);
    printf("parent is waiting for child to send SIGUSR1 at %s",
           ctime(&t));
    if (sigsuspend(&sigset) == -1)
        perror("sigsuspend() returned -1 as expected");
    time(&t);
    printf("sigsuspend is over at %s", ctime(&t));
}
}

```