ITT440 – NETWORK PROGRAMMING

Introduction To Unix Signals

Basic Knowledge

- Signals, to be short, are various notifications sent to a process in order to notify it of various "important" events.
- By their nature, they interrupt whatever the process is doing at this minute, and force it to handle them immediately.
- Each signal has an integer number that represents it (1, 2 and so on), as well as a symbolic name that is usually defined in the file /usr/include/signal.h or one of the files included by it directly or indirectly
  - HUP
  - INT
  - Etc
- Use the command 'kill -l' to see a list of signals supported by your system

Sending Signals Using The Keyboard

- **Ctrl-C**
  - Pressing this key causes the system to send an INT signal (SIGINT) to the running process. By default, this signal causes the process to immediately terminate.
- **Ctrl-Z**
  - Pressing this key causes the system to send a TSTP signal (SIGTSTP) to the running process. By default, this signal causes the process to suspend execution.
- **Ctrl-**
  - Pressing this key causes the system to send a ABRT signal (SIGABRT) to the running process. By default, this signal causes the process to immediately terminate. Note that this redundancy (i.e. Ctrl-\ doing the same as Ctrl-C) gives us some better flexibility. We'll explain that later on.

```c
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <signal.h>

int main(void)
{

    void sigint_handler(int sig); /* prototype */
    char s[200];

    if (signal(SIGINT, signal_handler) == SIG_ERR) {
        perror("signal");
        exit(1);
    }

    printf("Enter a string:\n");

    if (gets(s) == NULL)
        perror("gets");
    else
        printf("You entered: %s\n", s);

    return 0;
}

void sigint_handler(int sig)
{
    Printf("Not this time!\n");
}
```

Unix-signal.c

Example of SIGINT. Compile the program above and note the output.

Exercise

Write a program in C that will captured the following signal:

- ⊙ SIGKILL
- ⊙ SIGHUP
- ⊙ SIGTERM
  - ▪ When a signal is received, program will output: "This is a special signal handler for <signal>" substitute <signal> with the correct signal.

Source

http://neuron-ai.tuke.sk
/hudecm/Tutorials/C/special/signals/signals-
programming.html