

Homework 7

- You'll be writing three "programs": `hw7.c`, `hw7c1.c`, `hw7c2.c`
- Write a "Makefile" to compile and link each of the three programs
- The server (`hw7.c`) will fork two clients, and use an "exec()" variant to set up `hw7c1` and `hw7c2`.
- Client#2 (`hw7c2.c`) will pay attention to `stdin`.
 - If it receives a single number (ex. 10), it will send a signal with that number to Client#1.
 - If it receives a range (ex. 10-15), it will send every signal in that range to Client#1, sequentially, in increasing signal numbers.
 - If it receives a "q", it should terminate.
- The server will wait for Client#2 to terminate.
 - The server will then send a `SIGUSR2` to Client#1.
 - Client#1 should handle the signal by terminating, *but only if* the sending process was the server! (ie. use `sigaction()` to identify the sending process' pid).
 - When Client#1 terminates, the server will display a message "Server terminating".
 - The server will then terminate.
- The server should let its child processes know about pids they need to use by using the `argv` array when doing the `exec()`. Furthermore, the server should use a *path* style `exec()` call. These two constraints imply a specific form of `exec` you should use.
- Client#1
 - Client#1 will display a message such as "Rcvd signal %2d (give name of signal) from pid#number" for every signal it receives, identifying which process sent the signal (by its pid).
 - Client#1 should not terminate for any signal it receives other than `SIGUSR2` when its from the server, and, of course, `SIGKILL`.

hints: `strsignal()` `pause()`