* Signals
  + A signal is a small message that notifies a process that an event of some type has occurred in the system
  + The only information in a signal is its ID and the fact that it arrived
* Sending a signal
  + The kernel sends a signal to a destination process by updating some state in the context of the destination process
* Receiving a signal
  + A destination process receives a signal when it is forced by the kernel to react in some way to the delivery of the signal
* Pending and blocked signal
  + A signal is pending if sent but not yet received
  + There can be at most one pending signal of any particular type
  + Signals are not queued
  + A process can block the receipt of certain signals
  + Blocked signals can be delivered but will not be received until the signal is unblocked
  + Pending signals are received atmost once
* Pending/blocked bits
  + A Kernel maintains pending and blocked bit vectors in the context of each process
  + Pending - kernel sets bits in pending if the signal is delivered, kernel clears bits in pending if the signal is received
  + Blocked - can be set and cleared using sigprocmask function (signal mask)
* Process group
  + Every process belongs to exactly one process group
  + getpgrp() - return process group of current process
  + setpgid() - change process group of a process
  + /bin/kill - program sends an arbitrary signal to a process or process group
* Sending signals from the keyboard
  + Typing ctrl+c (ctrl+z) causes the kernel to send a SIGINT (SIGTSIP) to every job in the foreground process group
  + SIGINT - default action is to terminate each process
  + SIGTSTP - default action is to stop each process
* Pnb = list of all pending signals that are blocked
* Pnb = pending & ~blocked