

Signal handling

Most shells let you stop or pause processes with special keystrokes. These special keystrokes, such as `Ctrl-C` or `Ctrl-Z`, work by sending signals to the shell's subprocesses. For example, pressing `CTRL-C` sends the `SIGINT` signal, which usually stops the current program; pressing `CTRL-Z` sends the `SIGTSTP` signal, which usually sends the current program to the background.

Recall that your terminal window is running a shell program itself. The shell must make sure that these keystrokes do not stop the shell program itself. If you try these keystrokes in your homemade shell right now, the signals are sent directly to the shell process itself. This means that attempting to `CTRL-Z` a subprocess of your shell, for example, will also stop the shell itself. We want to have the signals affect only the subprocesses that our shell creates, even when your shell is idle.

TABLE OF CONTENTS

- [Example: shells in shells](#)
 - [Process groups](#)
 - [Foreground terminal](#)
 - [Overview of signals](#)
 - [Task](#)
-