

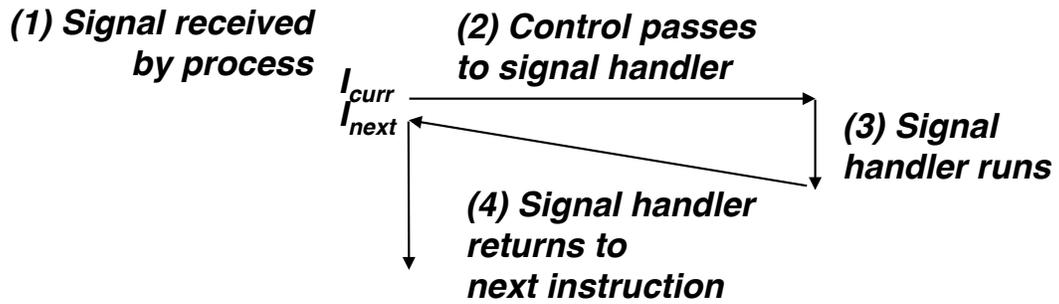
Shell Design

```
while (true) {  
    Print command prompt.  
    Read command line from user.  
    Parse command line.  
    If command is built-in, do it.  
    Else fork process to execute command.  
        in child:  
            Execute requested command with execv.  
                (never returns)  
        in parent:  
            Wait for child to complete.  
}
```

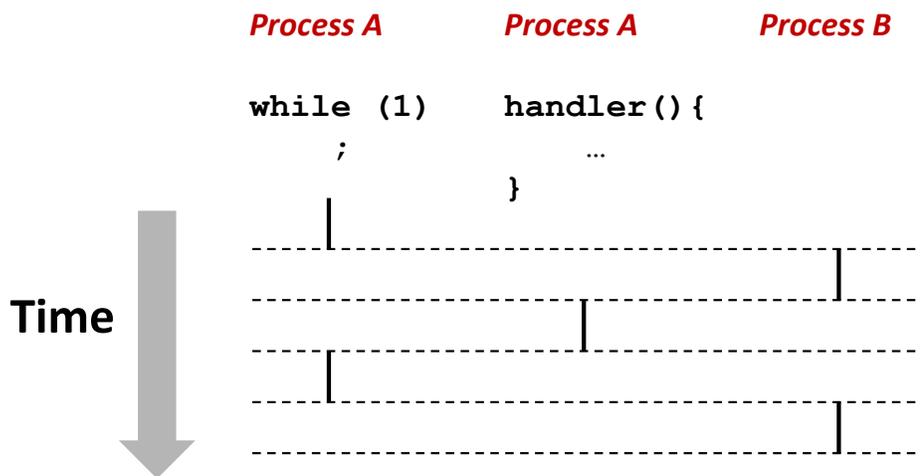
Signals

<i>ID</i>	<i>Name</i>	<i>Corresponding Event</i>	<i>Default Action</i>	<i>Can Override?</i>
2	SIGINT	Interrupt (Ctrl-C)	Terminate	Yes
9	SIGKILL	Kill process (immediately)	Terminate	No
11	SIGSEGV	Segmentation violation	Terminate & Dump	Yes
14	SIGALRM	Timer signal	Terminate	Yes
15	SIGTERM	Kill process (politely)	Terminate	Yes
17	SIGCHLD	Child stopped or terminated	Ignore	Yes
18	SIGCONT	Continue stopped process	Continue (Resume)	No
19	SIGSTOP	Stop process (immediately)	Stop (Suspend)	No
20	SIGTSTP	Stop process (politely)	Stop (Suspend)	Yes

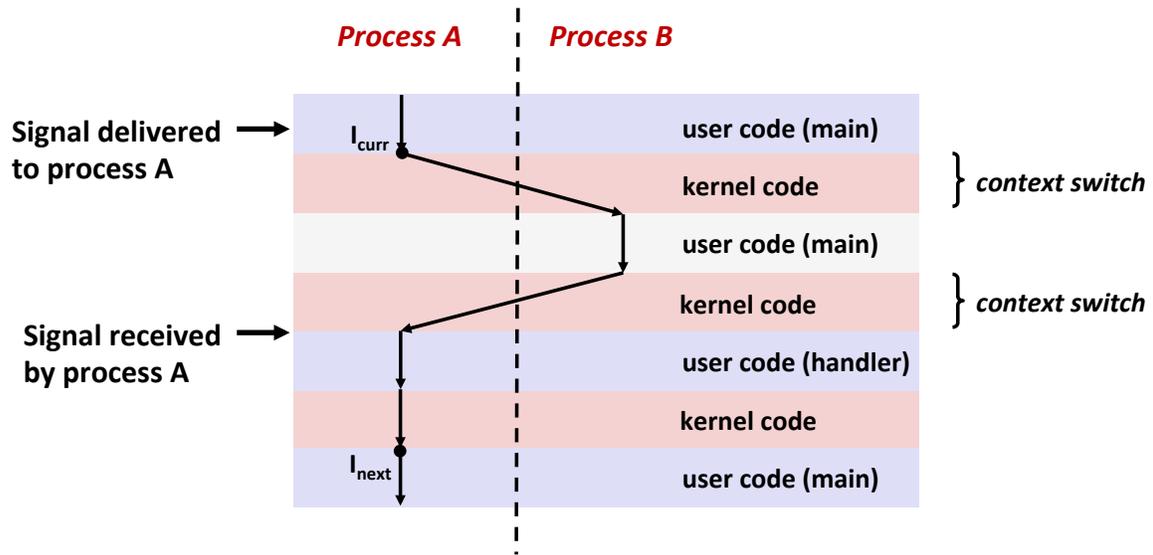
Signal Control Flow



Signal Handler as Concurrent Flow



Signal Handler as Concurrent Flow (alt)



Process Groups

