

Use the command `man ps` to get more information about the `ps` command. The task manager on Windows systems does not provide the parent process ID, but the *process monitor* tool, available from [technet.microsoft.com](http://technet.microsoft.com), provides a process-tree tool.

- 3.11 Explain the role of the `init` process on UNIX and Linux systems in regard to process termination.
- 3.12 Including the initial parent process, how many processes are created by the program shown in Figure 3.32?
- 3.13 Explain the circumstances under which the line of code marked `printf("LINE J")` in Figure 3.33 will be reached.
- 3.14 Using the program in Figure 3.34, identify the values of `pid` at lines A, B, C, and D. (Assume that the actual pids of the parent and child are 2600 and 2603, respectively.)

```
#include <sys/types.h>
#include <stdio.h>
#include <unistd.h>

int main()
{
    pid_t pid, pid1;

    /* fork a child process */
    pid = fork();

    if (pid < 0) { /* error occurred */
        fprintf(stderr, "Fork Failed");
        return 1;
    }
    else if (pid == 0) { /* child process */
        pid1 = getpid();
        printf("child: pid = %d",pid); /* A */
        printf("child: pid1 = %d",pid1); /* B */
    }
    else { /* parent process */
        pid1 = getpid();
        printf("parent: pid = %d",pid); /* C */
        printf("parent: pid1 = %d",pid1); /* D */
        wait(NULL);
    }

    return 0;
}
```

Figure 3.34 What are the pid values?