

Midterm 2–April 10, 1989

Open book section (36 points)

The exam is to be turned in at 2:50 pm. The closed book section should be turned in before you open your books and notes to work the open book section. For the open book section, write your answers on separate pieces of paper.

Problem 1. (21 points)

I log into two different terminals. On one terminal I start a program called `cpupig` in background. The shell responds with:

```
[1] 4828
```

informing me that `cpupig` is running with process ID 4828.

Immediately, from the other terminal I type: `printf("SIGBUS caught!\n");`

```
% kill -BUS 4828
```

which sends the SIGBUS (10) signal to process 4828. Now the first terminal, the one with `cpupig`, prints

```
[1] Bus error    cpupig
```

- (a) How did the shell know that `cpupig` was running as process 4828?
- (b) How did the shell know that `cpupig` was killed by a bus error when the signal was sent from another terminal?
- (c) Assuming `cpupig` does not handle bus errors, outline how the kernel sends the signal from `kill` to `cpupig`.

Now suppose `cpupig` is handling bus errors with the following handler:

```
sigbus_handler()  
{  
    kill(getpid(), SIGBUS);  
    printf("SIGBUS caught!\n");  
    exit(0);  
}
```

and that `cpupig` is running under a System V operating system like `napoleon` (thus, signals work as described in Bach's book).

- (d) Now, how does the kernel set up `cpupig` to receive the SIGBUS sent from the other terminal? Don't go into great detail. A picture and four or five sentences should do.
- (e) What happens when the `kill` system call within the handler is executed?

By the way, the system call `getpid` returns the process ID of the calling process, in this case 4828.

Problem 2. (15 points)

You, as a Unix guru, have been hired by a strange company. Your first task is to write a program called `cleanup` such that the command

`% cleanup file`

will delete *file* if and only if the user running `cleanup` has a user ID less than 1066 *and* *file* is owned by a user whose ID is greater than 1066.

How would you write such a program and install it in your system? Assume you know the superuser password.