

## OPERATING SYSTEMS

### 1st Homework

**Due on:** Friday, February 10th, 2006

**Submission:** Send your homework through the e-mail: [kurt@ce.itu.edu.tr](mailto:kurt@ce.itu.edu.tr)

**In this homework, you complete the following tasks:**

- \* Learn to execute command lines
- \* Learn how to handle background jobs

#### **Preparation**

- \* For Task-1, see `exec()` function and the `fork()` system call
- \* For Task-2, see `wait()` function

#### **Download**

<http://www.cs.itu.edu.tr/~kurt/Courses/os/hw1.tar.gz>

#### **Tasks**

Perform the following tasks:

1. Edit the `run_command.c` file so a child process is created to run the command, and the parent waits for the child process to terminate. Check for built in commands first; create a new process only for commands which are not built in. Use the parser to create from the command line the `argv` array passed to the child.

**Working directory:** `$HOME/hw1/task1`

**Files provided:** `Makefile`, `shell.h`, `shell.c`, `parse.c`, `builtin.c`, `run_command.c`

After the file has been edited, type **make** or **make shell**.

When you get the prompt, type **shell**.

The sample output is as follows:

```
./shell
Next? -> ls -l
total 56
-rw-r--r-- 1 kurtbin users2 2353 May 6 2003 builtin.c
-rw-r--r-- 1 kurtbin users2 1548 May 22 2005 builtin.o
-rw-r--r-- 1 kurtbin users2 266 May 22 2005 Makefile
-rw-r--r-- 1 kurtbin users2 51 May 6 2003 object_defs
-rw-r--r-- 1 kurtbin users2 1406 May 6 2003 parse.c
-rw-r--r-- 1 kurtbin users2 1192 May 22 2005 parse.o
-rw-r--r-- 1 kurtbin users2 1014 May 6 2003 run_command.c
-rw-r--r-- 1 kurtbin users2 1680 May 22 2005 run_command.o
-rwxr-xr-x 1 kurtbin users2 8468 Jan 27 10:55 shell
-rw-r--r-- 1 kurtbin users2 639 May 6 2003 shell.c
-rw-r--r-- 1 kurtbin users2 333 May 6 2003 shell.h
-rw-r--r-- 1 kurtbin users2 1308 May 22 2005 shell.o
Next? -> rm /tmp/trash
rm: cannot lstat `/tmp/trash': No such file or directory
Child 25360 exited with error status 1: Operation not permitted
Next? -> it has been snowing for 5 days
it: No such file or directory
Child 25589 exited with error status 2: No such file or directory
Next? -> quit
```

2. Edit the `is_background.c` file to detect an “&”. Alter the `run_command.c` file so that if a task is running in the background, the parent does not wait. Do not worry about background process becoming zombies at this point; this will be addressed later on the lectures.

**Working directory:** `$HOME/hw1/task2`

**Files provided:** `Makefile`, `shell.h`, `shell.c`, `parse.c`, `builtin.c`, `run_command.c`, `is_background.c`

After the file has been edited, type **make** or **make shell**.

When you get the prompt, type **shell**.

The sample output is as follows:

```
$/shell
Next? -> sleep 300 &
Next? -> ps
  PID TTY          TIME CMD
 20789 pts/12    00:00:00 bash
 26546 pts/12    00:00:00 shell
 26606 pts/12    00:00:00 sleep
 26620 pts/12    00:00:00 ps
Next? -> quit
```