

**CSCE 311, Section 001
Spring 2003**

Assignment #1

Assigned: 3 February 2003

Due: 17 February 2003

Objective: To familiarize students with the basics of the OSP simulator. Students will execute a working version of the simulator. Students should familiarize themselves with the contents of sections 1.1 through 1.4 of the OSP manual along with Chapter 2. While the information in these sections may not be needed to complete this assignment, the next assignment will follow shortly and will require some knowledge of the inner-workings of OSP that may not be covered directly in lecture.

Required to turn in: You must turn in a “writeup” including a table of the simulation results you are asked to provide in the following sections. This will be a hard copy, not an electronic copy. You should include any pertinent information from the steps of the project described below. This will not be the normal turn in procedure for OSP assignments. Future assignments will require that hard copy and electronic versions be handed in.

Follow these steps to become familiar with the OSP simulator:

1. Log into your CSE account from any CSE Sun workstation or remotely.

2. To see the OSP files set up for the demo from your directory:

```
% ls ~osp/demo/OSP.demo.sun4
```

To run the demo from your own directory:

```
% ~osp/demo/OSP.demo.sun4
```

This way of executing OSP requires you supply the system parameters interactively -- use the responses given in the following table:

Prompt	Response
Intensity of process creation	5
Average CPU time	80
Frequency of events	5
Memory-related events	1
I/O related events	5
Resource-related events	0
Socket type	s

CPU time quantum	20
Prepaging	0
Memory reference pattern	1 *
Handling deadlocks	a
Simulation time	45000
Snapshots	1

Trace switch	0
Interactive	n

*note this is the letter “ell” (l), not the number one (1)

Your responses will be recorded locally in the file `simulation.parameters`. Output will be directed to the file `simulation.run` as long as you answered “n” to the last question (interactive mode); otherwise results are printed on the screen. These two simulation file names are used over and over, so copy the information to another file whenever you want to keep it.

3. For subsequent runs, you can use the same parameters without having to type them in again. Copy `simulation.parameters` to a file called `demo1.parms`. The following command (typed from within your directory) will use the same parameters in a new run:

```
% cp simulation.parameters demo1.parms
% ~osp/demo/OSP.demo.sun4 demo1.parms
```

Save the file `simulation.run` as `demo1.run` so you can refer to it again. Examine the file to find values to complete the `demo1` column of this table.

Measured parameter	Demo1
System throughput	
Average number of processes during simulation	
Average CPU time per process	
Average waiting time per process	
Average turnaround time per process	

4. Copy the files `~osp/demo/demoX.parms` (where `X = 2,3,7`) to your local directory for use. Run simulations in exactly the same manner as previously using `demo2.parms`, `demo3.parms`. Save the resulting runs in `demo2.run` and `demo3.run` respectively. Separately run `demo1.parms`, `demo2.parms`, and `demo3.parms` a second time each. Save the corresponding results in files named `demo4.run`, `demo5.run`, and `demo6.run`. The result (including our earlier run) should be 6 files of simulator results in `demoX.run` (where `X=1,..,6`). Lastly, run the simulator using `demo7.parms` as input saving the run in `demo7.run` as well. Here is a chart summing up what we did in this step.

Input file	Output file
<code>demo2.parms</code>	<code>demo2.run</code>
<code>demo3.parms</code>	<code>demo3.run</code>
<code>demo1.parms</code>	<code>demo4.run</code>
<code>demo2.parms</code>	<code>demo5.run</code>
<code>demo3.parms</code>	<code>demo6.run</code>

demo7.parms

demo7.run

5. Add new columns for each demo (i.e. 2 through 7) to the chart you created in step3. Observe the following trying to answer the succeeding questions:

Compare demo1.run to demo4.run.

Compare demo2.run to demo5.run

Compare demo3.run to demo6.run

Compare demo1.parms, demo2.parms, demo3.parms

Compare demo7.parms to demo1.parms

Compare demo7.run to demo1.run

Why aren't the results of two runs with the same parameters identical?

What observations can you make regarding successive runs of the simulator using the same parameters?

What is the primary difference between demo1 and demo7?

How do you determine this?

What do the resulting runs tell you?

What changed between demos 1, 2, and 3?

How did this affect the simulation results?

6. Vary one more parameter of the simulator, call the output demo8, compare and include it in your chart results. What parameter did you vary? What does the parameter do? Can you determine how the resulting simulator run was affected? If so, then how. If not, then why not?