

CSC 2204F -- OPERATING SYSTEMS

UNIVERSITY OF TORONTO
FALL 2005

The TakeHome

Welcome to the 2204 take home. The main goal of the take home, as it was for most of the workload components of the course, is to have you learn something about operating systems.

The take home is available at a time tailored to each student's request. You should e-mail your answers, as an attachment in PostScript (ps) or Portable Document Format (pdf), to gsg@cs.utoronto.ca within two days (48 hours) after your start time.

There is no intent for you to take 48 straight hours of work to complete this assignment; there are 34 marks for the 48 hours. Keep a log of your hours spent and report it with your answers; this is for my curiosity only.

If you feel you need clarification, you may e-mail me (hopefully only once) on the first day (up to 10 p.m.) and you should have your answers back by the next morning.

And now, on with the show.

1. [8 marks]

The following software routines, written in a long-lost language, are being proposed as a solution to the two-thread critical section problem. **Prove** or **disprove**.

```

1  /* start global */
2  integer ask, inuse;
3  ask = 0;
4  inuse = 0;
5  /* end global */
6
7  /* the code for thread 1, T1, follows */
8
9  label1:
10     if ask != 0 then goto label1;
11     ask = 1;
12     if ask != 1 then goto label1;
13     if inuse != 0 then goto label1;
14     inuse = 1;
15     if ask != 1 then goto label1;
16
17     << critical section >>
18
19     ask = 0;
20     inuse = 0;
21     << rest of program >>
22     goto label1;
23
24
25  /* the code for thread 2, T2, follows */
26
27  label2:
28     if ask != 0 then goto label2;
29     ask = 2;
30     if ask != 2 then goto label2;
31     if inuse != 0 then goto label2;
32     inuse = 1;
33     if ask != 2 then goto label2;
34
35     << critical section >>
36
37     ask = 0;
38     inuse = 0;
39     << rest of program >>
40     goto label2;

```

2. [8 marks]

Read The Zax, by Dr. Seuss, available on-line. Although deadlock was not covered as a lecture topic in the course, that does not preclude me from asking about it here.

a) **Show** that the four necessary conditions for deadlock exist in The Zax. [Sometimes only three conditions are listed, but it is useful to consider that there are four necessary conditions.]

b) **Provide** an answer to (a), or **provide** an appropriate general commentary on the answer or the question, that is **in the style of Dr. Seuss**, and the marks that you receive for (a) may be doubled.

3. [8 marks]

Using English prose and getting technical only to the level of pseudo-code, **describe** in sufficient detail (so that I can determine that you know what you are talking about) how you would produce a graph showing the dependency of page fault rate on amount of memory for a process's pages for the Working Set model of page replacement. The vertical y-axis is labeled page fault rate, going from 0.0 to 1.0, and the horizontal x-axis is labeled fraction of process's pages (fraction of address space) in real memory, going from 0.0 to 1.0. The experiment that would produce such a graph involves processing a given page reference string, assuming an initially empty main memory, with a range of Working Set window sizes $\{\Delta_i\}$, $\Delta_i < \Delta_{i+1}$, and recording page faults.

4. [10 marks]

Consider preemptive priority processor scheduling of jobs in the following computer system. Jobs are either in the waiting set or in the running set. Priority values may be positive, negative, or zero; high priority in scheduling is afforded to a job with a high numeric priority value. All jobs with the highest priority value are in the running set simultaneously and are proceeding to completion receiving round-robin service; while in the running set, the priority value of each job there increases at rate B. All jobs not having the highest priority value are in the waiting set; while in the waiting set, the priority value of each job there increases at rate A. When a job initially arrives at the system, its initial priority value is 0.

In English, use a few sentences, and perhaps a diagram, each to **describe** the scheduling that results from the following choices of parameters:

- a) $B = 0$
- b) $B = A > 0$
- c) $B > A > 0$
- d) $0 > B > A$
- e) $A > B > 0$.

How many hours did you spend on the take home?

Which was the most difficult question for you?

2204 TakeHome