CS240
Advanced Topics in Operating Systems
Midterm – May 3, 2007

OPEN BOOK, OPEN NOTES

!! You should skip 10 points worth of questions !!

Your name: _____

Stanford ID: _____

In accordance with both the letter and the spirit of the Stanford Honor Code, I did not cheat on this exam. Furthermore, I did not and will not assist anyone else in cheating on this exam.

Signature: _____

The quiz has 15 questions. You have 75 minutes to complete the remainder. Some questions may be much harder than others.

I Speculative Execution

1. [**5 points**]**:** What optimizations is xsyncfs doing in Figure 1? What is the difference between eager and output triggered commits in xsyncfs? How is eager commit xsyncfs different from just making ext3 synchronous?

2. [**5 points**]**:** What different things does xsyncfs do when `exit` is called? Why? Give two example uses that would cause it to switch between these behaviors.

II Boehm tricks

3. [5 points]: Consider the code:

```
initialization:
    int y = 0;
    int x = 0;

  thread  1   |  thread 2
  ----------------------
  1: r1 = x;  | 3: r2 = y;
  2: y =  1;  | 4: x = 2;
  ------------------------
```

Give all values you could see for `r1` and `r2` after running this code, explaining anything surprising.

III Eraser

4. [10 points]: You have a structure:

```
struct foo {
        // a and b are of size= 1 bit.
        unsigned a:1,
                 b:1;
        lock a_lock;   // protect's field a.
        lock b_lock;   // protect's field b.
};
```

And your program declares a global `f` of type `struct foo`. It then runs this code in thread T1:

```
lock(f.a_lock);
f.a = 1;
unlock(f.a_lock);
```

And this code in thread T2:

```
lock(f.b_lock);
f.b = 1;
unlock(f.b_lock);
```

What error(s) will Eraser flag? Are these error(s) real or false positives? What if you change `a` and `b` to be of type `char`?

IV Mesa

5. [5 points]: Your code has two monitors, M1:

```
entry foo1() { bar1(); }
entry foo2() { bar2(); }
```

And M2:

```
condition c;
entry bar1() { wait(c); }
entry bar2() { signal(c); } }
```

Your program calls into M1. What can happen? Why does MESA work this way compared to the alternative?

6. [5 points]: Eraser claims that: "Monitors are an effective way to avoid data races if all shared variables are static globals, but they don't protect against data races in programs with dynamically allocated shared variables..."

Is this claim correct?

V The world according to Burrows

7. **[10 points]:** Which statement is not true and why?

A. Event-driven code *always* rips apart functions, spreading one logical task over a number of functions.

B. Event-driven code typically cannot take advantage of multiprocessors.

C. Event-driven programs block on a page fault.

D. They don't use Eraser at Google because it had too many false positives.

VI VMS

8. [5 points]: In terms of virtual memory stuff: what do you have to do on context-switch?

9. [5 points]: What happens if the RSS limit is too high? Too low? What happens if the difference between the high and low watermark on the modified list is too small? Too large?

VII Appel and Li

10. [5 points]: Which is more expensive, `prot1` or `unprot`? Why? If `protN` is useful, why not also provide `unprotN`?

11. [5 points]: Assume your OS does not provide `DIRTY`: how to emulate it at user level?

VIII Superpages

12. [10 points]:

In the paper "Practical, transparent operating system support for superpages," by Navarro et al., the authors propose some virtual memory implementation changes that generally benefit application performance. However, there is always the danger that an OS change that benefits one application may harm another. Which of the following properties of the proposed scheme could potentially hurt performance?

Circle either True or False next to each statement. (2 points each)

A. **True** **False** The author's implementation uses an expensive SHA-1 hash function to detect which subpages of a superpage have been modified. Computing SHA-1 consumes CPU time that would otherwise be available to user applications.

B. **True** **False** Paging superpages to disk must cause fragmentation in the swap partition, thereby increasing seek times during page faults.

C. **True** **False** Superpage support can cause more cache misses in a direct-mapped, physically-addressed cache if many processes use the same virtual address for different physical memory.

D. **True** **False** Under memory pressure, the superpage scheme no longer evicts pages in (approximate) least-recently-used order, which can cause an increase in paging activity.

E. **True** **False** The proposed scheme uses extra pages of physical memory for reservations, which can result in applications being paged out even when there are a fair number of unused pages of physical memory.

IX VMMs

For each of the following workloads, explain whether you would expect better performance with a software or hardware VMM similar to the ones described in the Adams and Agesen ASPLOS paper. Justify your answers with concrete data from the paper.

13. [10 points]:

- A userlevel program the iteratively computes the first 100000 Fibonacci numbers.

- 100 invocations of a C program that reads /etc/passwd a line at a time and printf()'s each one with any uppercase letters converted to lowercase and vice versa.

- A network daemon that each time it gets a connection fork()'s a child process that reads in a paragraph of text from the network a line a time, printf()'s each one with any uppercase letters converted to lowercase and vice versa, and then exit()'s.

X ESX

14. [**5 points**]**:** One problem with a hypervisor implementing transparent paging is the "double paging." Imagine an OS that implements transparent page sharing (just internally, for its kernel and its applications). Is there a similar "double sharing" problem? Why or why not?

15. [**5 points**]**:** Most OSes have some sort of `mpin(void *va, unsigned len)` system call, which will pin a range of addresses in physical memory. Explain how to use this system call to replace the need to write a balloon driver. What is the advantage/disadvantage of this approach?