
Name:

CIS 586 Midterm

March 6th, 2008

Answer all questions in the space provided. You may write on the back if you need more room. Closed book, closed notes, closed neighbor. Please write your name on each page.

Question 1 /10 pts

Question 2 /5 pts

Question 3 /10 pts

Question 4 /15 pts

Question 5 /15 pts

Question 6 /15 pts

Question 7 /20 pts

Question 8 /10 pts

Total / pts

Question 1) a) (5) Explain the difference between **user mode** and **kernel mode**.

b) (5) How does the system switch from **user mode** to **kernel mode**.

Question 2) (5) Why do all operating systems (that run on real hardware) require assembly language routines.

Question 3) (10) What are the main disadvantages of user level threads?

Question 4) (15) The below code synchronizes cars passing over a bridge. The bridge is only wide enough for one-way traffic, and can only support 5 cars a time. While traffic is passing from one-side the traffic on the other side has to wait.

```
Semaphore mutex = new Semaphore(1);
Semaphore load = new Semaphore(5);
Semaphore bridge = new Semaphore(1);
Integer east_count = 0;
Integer west_count = 0;
```

Car East thread

```
mutex.P();
east_count = east_count + 1;
if east_count == 1 then
    bridge.P();
end if
mutex.V();
load.V();
bathroom code
load.P();
mutex.P();
east_count = east_count - 1;
if east_count == 0 then
    bridge.V();
end if
mutex.V();
```

Car West thread

```
mutex.P();
west_count = west_count + 1;
if west_count == 1 then
    bridge.P();
end if
mutex.V();
load.P();
bathroom code
load.V();
mutex.P();
west_count = west_count - 1;
if west_count == 0 then
    bridge.V();
end if
mutex.V();
```

Are there any synchronization issues with this code? Does it successfully synchronize cars as described above? (We are not asking for any improvements, simply point out any errors.)

(intentionally blank)

This study resource was
shared via CourseHero.com

Question 5) (10) What are the main differences between monitors and semaphores?

Question 6) (15) Explain what is meant by feedback in a multi-level feedback scheduler. Is a multi-level feedback scheduler a good scheduler for an interactive system?

Question 7) **(Read the entire question before answering)**

(20) There are 5 processes and 5 resources, resources are currently allocated according to:

$$C = \begin{pmatrix} 2 & 1 & 1 & 1 & 1 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$E = \langle 4, 5, 5, 4, 3 \rangle$$

$$A = \langle 0, 1, 0, 2, 1 \rangle$$

The resources each process still need to finish is given by:

$$R = \begin{pmatrix} 2 & 2 & 3 & 0 & 2 \\ 1 & 1 & 0 & 2 & 0 \\ 2 & 0 & 3 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 2 & 0 \end{pmatrix}$$

The system is in state above and process 0 requests 1 unit of the 5th resource (first row, last column).

- (a) (15) If we are using deadlock avoidance with the banker's algorithm, should this request be serviced? **HINT:** You first need to update the state of the system with the request.

- (b) (5) What does the banker's algorithm tell us about the request?

Question 8) (10) Why do all operating systems use the Ostrich approach rather than one of the three other approaches? (i.e. What are the **major** hindrances to the other three approaches?)

This study resource was
shared via CourseHero.com