

# MCS378 Midterm Exam

Serial #:

This exam is an open-book, open-notes, individual-work exam. Please write your name only on this page. Do not turn the page until instructed, in order that everyone may have the same time. Then, be sure to look at all problems before deciding which one to do first. Some problems are easier than others, so plan your time accordingly. You have 50 minutes to work.

**I am willing to give hints; come see me if you need one.**

Write the answer to each problem on the page on which that problem appears. You may use the back of that same sheet if you need additional space. You may also request additional paper.

Name: \_\_\_\_\_

Problem	Page	Possible	Score
1	2	12	
2	3	12	
3	4	12	
<b>Total</b>		36	

1. [ **12 Points** ] Suppose that in order to prevent chaos at my house, I establish a rule that there can never be more than twice as many children as adults present. Note this restricts both the entrance of children and the exit of adults: if we have four children and two adults, we can neither let a fifth child in nor let one of the adults leave. In order to enforce this rule, I could use a semaphore, `capacity`, which represents the number of additional children that we have room for. Each adult is required to execute the `adultEnter` procedure after entering the house and the `adultExit` procedure before leaving. Similarly, each child must execute the `childEnter` procedure before entering and the `childExit` procedure after leaving. These procedure operate on the `capacity` semaphore. In particular, the `childEnter` and `adultExit` procedures won't return until it is legal to enter or exit. In the following pseudo-code, I use `down(capacity)` and `up(capacity)` for the semaphore operations, rather than the book's P and V, respectively.

```

adultEnter:                adultExit:
  up(capacity)             down(capacity)
  up(capacity)             down(capacity)

childEnter:                childExit:
  down(capacity)           up(capacity)

```

The above operations suffice to keep my house always in a legal state, with never more than twice as many children as adults. None the less, it would be preferable to add a second semaphore, `mutex`, initialized to 1, and change the `adultExit` procedure to the following:

```

adultExit:
  down(mutex)
  down(capacity)
  down(capacity)
  up(mutex)

```

Explain why this second version is preferable to the first. In particular, give a scenario where something undesirable can happen with the first version.

2. [ **12 Points** ] Suppose a system has two page frames, initially both empty. Construct a reference string for which the FIFO page replacement policy results in a different number of page faults than the LRU page replacement policy. For each of the two page replacement policies, show for each page number in your reference string whether it is a hit (no page fault) or a miss (page fault), and if a miss, what pages are now in the frames.

3. [ **12 Points** ] Write a paragraph briefly summarizing one of the papers that we read for a student-led class. It may not be a paper for which you led the class. Your summary should state what problem the authors were trying to solve, and what the general nature of their solution is.