NAME: _____ **(First name first) TOTAL:** ____

COSC 4330 FINAL DECEMBER 17, 1998

Closed Book. You can have ***one page*** of notes.

1. **True or False:** (5×2 points)

   T ___ **F** **_X_** Peterson's algorithm requires the use of a test-and-set instruction.

   T ___ **F** **_X_** Semaphore variables have no value.

   T **_X_** **F** ___ When a monitor procedure issues a signal on a monitor condition and another procedure catches that signal , it immediately releases the monitor.

   T ___ **F** **_X_** Memory compaction is used to fight internal fragmentation.

   T ___ **F** **_X_** It makes no sense for a virtual memory system to have a valid bit and a dirty bit.

2. Complete the following sentences: (6×5 points)

   In some implementations of monitors the signal operation is replaced by a

   _**notify**_____ operation.

   The page table organization where there is one page table entry per page frame is called

   _**inverted page table**_____.

   The cost of a TLB miss handled by the system's microcode is __**one extra memory**

   reference___.

   The page replacement policy of Berkeley UNIX simulates a non-existent _page

   refererenced_or_use___ bit using the _**valid**_____ bit of the machine.

   The file organization that provides the fastest access to the file data is _**sequential**

   (or contiguous)_____.

3. A computer has 64 bit addresses and a page size of eight kilobytes.(2×5 points**)**

a. How many bits are used to represent the page number? _51_ **bits**

b. What is the maximum number of entries in a process page table? _2^{51}_ **bits**

(Since the page size is 8KB, the last $\log_2 8{,}192 = 13$ bits of the address form the byte offset. The remaining 64 -13 bits of the address become the page number.)

4. Name one page replacement policy (2×5 points)

a. that handles well real time processes: _Windows/VMS_____

b. that cannot be implemented efficiently: _LRU or Working Set_____

5. Given the page reference string:

0 0 6 0 2 0 2 2 3 3 6 0 5 6 3 1 5 2 0

and a memory size of ***four*** page frames, list the pages that would be expelled from main memory:

a. under a LRU policy (5 points): _2, 0, 6 and 3 (in that order)_____

b. under a FIFO policy (5 points): _0, 6 and 2 (in that order)_____

6. The block size of a **32-*bit*** Berkeley UNIX file system is 8 kilobytes. How many file ***blocks*** can be accessed:

a. directly from the i-node? (2 points) _12_____ **blocks**

b. with one level of indirection? (3 points) _2^{11} or 2^{11}+12_____ **blocks**

c. with two levels of indirection? (5 points) _2^{19}-2^{11}-12 or 2^19___ **blocks**

7. Consider a producer-consumer relationship where there are two producers each producing a part of a complete item (say, the left shoe and the right shoe of a pair of shoes) and a consumer that can only consume complete items (most of us prefer to buy a pair of shoes than an individual shoe).

Add the missing semaphores to the following solution of the problem. (You can assume that all shoes are of the same model, the same color and the same size.)

```
#define NSLOTS ... // number of pairs the buffer can handle
semaphore mutex = 1;
semaphore left_slots = NSLOTS;
semaphore right_slots = NSLOTS;
semaphore left_shoes = 0;
semaphore right_shoes = 0;

make_left_shoe()  {
  struct left_shoe item;
  for(;;)
    produce(&item);
    P(&left_slots);
    P(&mutex);
    put(item);
    V(&mutex);
    V(&left_shoe);
  }
}; // make_left_shoe

make_right_shoe() {
        struct right_shoe item;
        for(;;) {
          produce(&item);
          P(&right_slots);
          P(&mutex);
          put(item);
          V(&right_shoe);
          V(&mutex);
        }
}; // make_right_shoe

consumer() {

  struct left_shoe one;

  struct right_shoe two;

  for(;;) {

    P(&left shoes); P(&right shoes); P(&mutex);   // SIMPLEST SOLUTION

    take(one);

    ____________________________________

    take(two);

    V(&mutex); V(&left slots); V(&right slots);

    wear_shoes();

} // consumer
```