### Exam 2

Name ____________________________________________

<table>
<thead>
<tr>
<th>Problem</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
<td></td>
</tr>
</tbody>
</table>
Directions:

0) Closed book exam. One 8.5"x11" summary sheet is permitted.

1) Place your initials and page number in the top, right hand corner of every page and your name on the cover page of the test.

2) Provide sufficient explanation so that I can follow your solution.

3) Problems are not given in the order of difficulty. Point values for each problem are indicated in parentheses. Spend your time accordingly.

Problem 1 (10 Points)

Three processes share four resource units that can be reserved and released only one at a time. Each process acquires and releases resources over time but only needs a maximum of two resource units at any one time. Deadlock can not occur. Why?

Problem 2 (10 Points)

a) What is meant by the terms internal memory fragmentation and external memory fragmentation of a dynamic memory allocation algorithm?

b) What are the properties of a good dynamic memory allocation algorithm?

Problem 3 (10 Points)

Consider a paged virtual memory system with its single-level page table stored entirely in main memory.

a) If a memory reference takes 100 nanoseconds, how long does a paged memory reference take when there is no TLB and the referenced page is in main memory?

b) If the system has a TLB and 80% of all page table references are found in the TLB, what is the effective virtual memory access time if it takes zero time to search the TLB?

Problem 4 (10 Points)

Consider the following virtual page reference stream:

\[ 0, 1, 2, 3, 0, 1, 4, 0, 1, 2 \]

a) Assume that a process is allocated three page frames. Show the behavior of the basic clock algorithm for the first FIVE page references by showing the evolution of the page frame assignments for these references.

b) What is the working set \( W(t = 9, \Delta = 4) \); i.e., tick 9 (counting from 1) and a window size of 4?
**Problem 5** (10 Points)

Suppose that the virtual page reference stream contains repetitions of long sequences of page references followed occasionally by random page references. For example, the sequence "0, 1, ..., 511, 431, 0, 1, ..., 511, 332, 0, 1, ..." consists of repetitions of the sequence "0, 1, ..., 511" followed by a random reference to pages 431 and 332.

a) Why won't the standard replacement algorithms (LRU, FIFO, Clock) be effective in handling this workload for a page allocation that is less than the sequence length?

b) If this program were allocated 500 page frames, describe a page replacement approach that would perform much better than the LRU, FIFO, or Clock algorithms.

**Problem 6** (10 Points)

Consider the following two-dimensional array:

```c
int X[64][64];
```

Suppose that a system has 4 page frames and each frame is 128 words (an integer occupies one word). Programs that manipulate the X array fit into exactly one page and always occupy page 0. The data is swapped in and out of the other 3 frames. The X array is stored in row-major order (i.e., X[0][1] follows X[0][0] in memory).

Which of the two code fragments shown below will generate the lowest number of page faults? Explain and compute the total number of page faults.

**Fragment A**

```c
for (int j = 0; j < 64; j++)
    for (int i = 0; i < 64; i++) { X[i][j] = 0; } 
```

**Fragment B**

```c
for (int i = 0; i < 64; i++)
    for (int j = 0; j < 64; j++) { X[i][j] = 0; } 
```

**Problem 7** (10 Points)

Most time-sharing systems use round-robin scheduling within priority levels where a process' priority is periodically adjusted based on its CPU utilization. Explain how this approach achieves fairness, low average response time, and/or high throughput?

**Problem 8** (10 Points)

Under high load, the disk request throughput of the LOOK disk scheduling algorithm will be substantially higher than the throughput for the FCFS algorithm but introduces the possibility for starvation. Describe AND justify a simple modification to the LOOK algorithm that has the following combined properties:

1) Prevent the starvation of disk requests.

2) Provide disk request throughput much higher than FCFS.
Problem 9 (10 Points)

A typical RAID subsystem consists of multiple drives sitting on a single bus managed by controller. If we assume the following:

- There are 15 drives that rotate at 10,000 RPM;
- Strips are 8 KB;
- The formatted track capacity is 64 KB; and

What is the minimum time required to transfer 15 strips if all drives must seek 20 msec? Explain.

Problem 10 (10 Points)

One of the benefits claimed by systems that support threads is that multithreaded programs can run faster (i.e., finish sooner) than single-threaded programs. What property (or properties) must a program have in order for this statement to be true when the multithreaded system is implemented by mapping each user-level thread to its own kernel-level thread (a 1:1 model)? Explain.