Final Exam – Graduate Operating Systems
CSE 60641 – Fall 2019

NAME: ________________________________________________________________

Reminders

1. Make sure to look through entire exam from front to back.
2. Please be concise in your answers if possible. A dissertation response is not needed for most questions.
3. Illustrations can be helpful to convey your point.
4. When in doubt, ask a question.
5. Try to at least have a partial answer for each exam item.
6. Many answers depend on context. Clearly define the context if necessary for your answer.
7. The exam is closed book, closed note, closed computer, closed electronic device!
8. Write your answers legibly.
**Question 1: Thrashing** occurs when more memory is being actively utilized than the system contains. When talking about thrashing, one often refers to the working set of a process, i.e., the pages that a process currently needs. The table below shows seven processes, how many frames in memory they each occupy, their working set sizes, and their current page fault rates (which is the number of page faults divided by the number of memory accesses). Given this information, answer the following questions:

   a.) Which process experiences the most thrashing? (Explain!)

   b.) Assume you wanted to eliminate thrashing for process P4, how many frames would you have to allocate? Explain!

   c.) For the scenario in b.), you need to select a “victim process” that will provide these frames to P4. Which victim process would you select and why?

   d.) Does this approach of adjusting allocations represent a global or local allocation scheme? Explain!

   e.) If you were able to allocate additional frames to the processes in the table, how many additional frames would you need in total to eliminate thrashing? Explain!

<table>
<thead>
<tr>
<th>Process</th>
<th>Frames Allocated</th>
<th>Working Set Size</th>
<th>Page Fault Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>50</td>
<td>60</td>
<td>0.1</td>
</tr>
<tr>
<td>P2</td>
<td>70</td>
<td>120</td>
<td>0.5</td>
</tr>
<tr>
<td>P3</td>
<td>50</td>
<td>70</td>
<td>0.2</td>
</tr>
<tr>
<td>P4</td>
<td>20</td>
<td>30</td>
<td>0.2</td>
</tr>
<tr>
<td>P5</td>
<td>150</td>
<td>165</td>
<td>0.2</td>
</tr>
<tr>
<td>P6</td>
<td>120</td>
<td>120</td>
<td>0.0</td>
</tr>
<tr>
<td>P7</td>
<td>45</td>
<td>75</td>
<td>0.4</td>
</tr>
</tbody>
</table>
**Question 2:** Answer the following questions about virtualization.

a.) What is the difference between VMM and VM?

b.) What does it mean for an architecture to be **virtualizable**? (Hint: explain this using the terms “sensitive” and “privileged”)

c.) Clearly explain the difference between a **host OS** and a **guest OS** in a virtual machine system.

d.) How does **paravirtualization** fix the problem of the non-virtualizability of x86 computers?

e.) How does **dynamic binary translation** fix the problem of the non-virtualizability of x86 computers?

f.) What makes virtualization “fast” in **fast dynamic binary translation**?

g.) What is the difference between a **system VM** and a **process VM**?

h.) What is a main advantage/reason to use a high-level-language VM (e.g., Java VM)? (Hint: it has nothing to do with performance)
**Question 3:** Consider a periodic real-time task set consisting of tasks $T1(2,10)$, $T2(2,8)$, $T3(1,5)$, and $T4(1,4)$, where the first number in parenthesis is the **worst-case execution time** and the second number is the **period**. Assume that the deadline is the end of the period. Answer the following questions:

a.) What is each task’s utilization and the total CPU utilization?

b.) In a system that uses EDF as the scheduling policy, is this task set schedulable? (Justify your answer!)

c.) In a system that uses rate-monotonic (RM) scheduling, can we guarantee that the tasks will always meet their deadlines? (Justify your answer!)

d.) In a system that uses rate-monotonic scheduling, is task T1 or task T3 more likely to miss deadlines? (Justify your answer)

e.) Assume that each task releases an instance (job) at time = 0. Draw a Gantt chart below that shows the schedule of this task set for 20 time units when an EDF scheduler is used. Clearly indicate each job’s deadline (which is the same as the period and release time). Also clearly indicate when a deadline is missed.
f.) Assume that each task releases an instance (job) at time = 0. Draw a Gantt chart below that shows the schedule of this task set for 20 time units when a preemptive RM scheduler is used. Clearly indicate each job's deadline (which is the same as the period and release time). Also clearly indicate when a deadline is missed.

**Question 4:** Answer the following questions about TLB.

a.) Explain what a **TLB flush** is.

b.) How does a TLB improve system performance?

c.) In a single-processor system, the TLB can either be shared among all runnable processes or dedicated to only the currently running process. Name an advantage of either approach.

d.) In a multi-processor system, each processor will typically have its own TLB. Why? (In other words: why do the processors not share the same TLB?)
Question 5: Assume that we use a DVS algorithm to decide on a processor frequency for the task set shown below. Assume that the beginning of the periods of all tasks coincide at t = 0. The processor supports 1GHz, 800MHz, 600MHz, 400MHz, and 200MHz. All execution times in the table are for the maximum frequency. Note that the actual execution time for a job is not known until it has completed execution. As discussed in class, the execution time is inversely proportional to the frequency. Assume that the system uses EDF scheduling.

a.) Given the task set below, what is the assumed utilization at t = 0? Is the task set schedulable at the maximum frequency? (explain)?

b.) Using static voltage scaling, at which frequency will the processor operate? Explain your answer!

c.) Using the cycle-conserving approach discussed in class, at which frequency will the processor operate at each of the following time instants (explain your answers):

- t = 0:

- t = 1:

- t = 4:

- t = 5:

- t = 10:

<table>
<thead>
<tr>
<th>Process</th>
<th>Max. Execution Time</th>
<th>Actual Execution Time</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>P2</td>
<td>4</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>P3</td>
<td>2</td>
<td>2</td>
<td>20</td>
</tr>
</tbody>
</table>