**Laboratory Quiz 2 – 2021.11.14.09.30.00**

I scored 9/10 in this laboratory quiz. However, not all questions are equal weighting. 5 questions each worth 1.5 scores, another 5 questions worth 1.0 score. The total is out of 12.5. I scored 11.5/12.5 overall. That means I probably have got it wrong on a question that worth 1.0 score.

**Green color highlight** denotes my selected answers which I believe are correct. Red color highlight denotes my selected answer which I believe I got it wrong. Take my guess with a grain of salt because I am also unsure what the final answer is.

**Q1: Given the following code:**

Inc1->Fork(Inc\_Consistent,0,1);

Inc2->Fork(Inc\_Consistent,1,1);

Dec1->Fork(Dec\_Consistent,2,1);

Dec2->Fork(Dec\_Consistent,3,1);

currentThread->Join(Inc1);

currentThread->Join(Dec1);

currentThread->Join(Inc2);

currentThread->Join(Dec2);

if(value==0)

printf("congratulations! passed.\n");

else

printf("value=%d, failed.\n", value);

**At the line “if(value==0)”, what will happen?**

* **The main thread wait for all threads to finish their operations and the value updated based on operations done by all four threads.**
* The main thread wait for Dec1 and Dec2 to finish their operations and the value updated based on operations done by the two threads.
* The main thread wait for Inc1 and Inc2 to finish their operations and the value updated based on operations done by the two threads.
* The main thread does not wait for any of the 4 threads.

**Q2: Page fault occurs in NachOS when a matching entry cannot be found in:**

* TLB
* **IPT (memory table)**

**Q3: There is no race condition problem if all the processes perform read-only operations on shared memory.**

* **True**
* **False**

I selected True in the exam, but I think it should be False.

**Q4: When a thread is in its critical section with a semaphore system, which of the following statements need to be executed by other threads to guarantee mutual exclusion in NachOS?**

* **mySem->P()**
* mySem->Release()
* mySem->V()
* mySem->Acquire()

**Q5: Given thread t and semaphore S, which of the following operations in NachOS may cause context switch?**

* t->Join()
* S->V()
* **t->Fork()**

**Q6: Page replacement procedure lruAlgorithm() is called in:**

* InsertToTLB()
* **PageOutPageIn()**
* VpnToPhyPage()

**Q7: By calling inertToTLB function, new entry will be inserted in cell which has valid flag equals to True.**

* True
* **False**

**Q8: Consider the LRU algorithm**

int phyPage = 0;

for (int i=0; i<**A**; i++) {

if (**B**)

return i;

}

for (int i=0; i<**A**; i++) {

if (**C**) {

phyPage = i;

}

}

return phyPage;

**What are A, B, C?**

* **A: NumPhysPages  
  B: !machine->tlb[i].valid  
  C: memoryTable[phyPage].lastUsed > memoryTable[i].lastUsed**
* A: NumPhysPages  
  B: machine->tlb[i].valid  
  C: memoryTable[phyPage].lastUsed > memoryTable[i].lastUsed
* A: TLBSize  
  B: !machine->tlb[i].valid  
  C: memoryTable[phyPage].lastUsed > memoryTable[i].lastUsed
* A: TLBSize  
  B: machine->tlb[i].valid  
  C: memoryTable[phyPage].lastUsed > memoryTable[i].lastUsed

**Q9: Which of the following statement is incorrect for NachOS?**

* Semaphore value can never be negative
* Semaphore is implemented using blocking
* **Semaphore is implemented using busy waiting**

**Q10: When there is a TLB miss, there must be a page fault.**

* True
* **False**