·TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

# Fall 2022 COMP 3511 Homework Assignment 3 (HW3)

**Handout Date: October 28, 2022, Due Date: November 11, 2022**

| | |
|---|---|
| Name | Casper kristiansson |
| Student ID | 20938643 |
| ITSC email | cok@connect.ust.hk |

**Please read the following instructions carefully before answering the questions:**

- You should finish the homework assignment **individually**.
- When you write your answers, please try to be precise and concise.
- **Homework Submission:** submitted to **Homework #3** on **Canvas**.
- TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

**1. [20 points] Multiple choices.**

Please write down your answers in the boxes below:

| Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 |
|---|---|---|---|---|---|---|---|---|---|
| **C** | **B** | **C** | **D** | **A** | **A** | **B** | **C** | **B** | **D** |

1) Which of the following statements on rate-monotonic (RM) scheduling algorithm is NOT correct?

A. It deals with periodic jobs

B. It is optimal under static-priority policies

C. The priority of a process is determined by the inverse of the length of period

D. All of the above

2) What would be the problem of the following code?

```
semaphores initialization;
signal (semaphore1);
critical section code;
signal (semaphore1);
```

A. Busy waiting

B. Mutual exclusion violation

C. Starvation or indefinite blocking

D. Deadlock

3) What is the purpose of the counting semaphore full in the implementation of the bounded-buffer problem using semaphores? (Slide 7.3-7.5)

A. It is used as a counter for counting the number of items in the buffer

B. It is used similarly as a condition variable specifying when full = 0, the cosumer process has to wait

C. Both A and B

D. Nove of the above

4) Which of the following statements on adaptive mutex used in Solaris is TRUE?

A. It is used for protecting data from short code segments and starts as a a spinlock in a multiprocessor system

B. If the lock held, and by a thread running on another CPU, it spins to wait for the lock to become available

C. If the lock held by a *non-run-state* thread, it blocks and sleeps waiting for signal of lock being released

D. All of the above

5) Assume there are three resources, R1, R2, and R3, that are each assigned unique integer values 35, 20, and 15, respectively. What is a resource ordering which prevents a circular wait, thus a deadlock?

A. R3, R2, R1

B. R1, R2, R3

C. R3, R1, R2

D. R2, R1, R3

6) Suppose that there are 12 resources available to three processes. At time 0, the following data is collected. The table indicates the process, the maximum number of resources needed by the process, and the number of resources currently owned by each process. Which of the following process sequence satisfies the safety criteria?

| Process | Maximum Needs | Currently Owned |
|---|---|---|
| $P_0$ | 10 | 4 |
| $P_1$ | 3 | 2 |
| $P_2$ | 7 | 4 |

A. P1, P2, P0

B. P2, P0, P1

C. P0, P2, P1

D. It is not safe.

7) Given the logical address 0xAEF9 (in hexadecimal) with a page size of 256 bytes, what is the page number and page offset?

A. 0xA and 0x0EF0

B. 0xAE and 0xF9

C. 0xAEF and 0x0009

·TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

D. 0xF9 and 0xAE

8) Consider a logical address space of 32 pages with 1024 bytes per page, how many bits are required in the logical address?

A. 13 bits

B. 14 bits

C. 15 bits

D. 16 bits.

9) Consider the following segment table:

| **Segment No.** | **Base** | **Length** |
|---|---|---|
| 0 | 1320 | 600 |
| 1 | 2020 | 356 |
| 2 | 40 | 120 |
| 3 | 3108 | 2304 |

Which of the following logical address will produce trap addressing error?

A. 0, 430
B. 1, 398
C. 2, 100
D. 3, 632

10) If memory access time is 250 nanoseconds, TLB access time is 20 nanoseconds and TLB hit ratio is 0.9, what is the effective memory access time?

A. 285

B. 315

C. 305

D. 295

**2. [20 points] Synchronization**

In the following code, three processes produce output using the routine putc and synchronize using two semaphores L and R.

```
/* Initialize the two semaphores */
semaphore L = 4;
```

·TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

semaphore R = 0;

| /* Process 1 */<br>while (true) {<br>wait(L);<br>putc('C');<br>signal(R);<br>} | /* Process 2 */<br>while (true) {<br>wait(R);<br>putc('A');<br>putc('B');<br>signal(R);<br>} | /* Process 3 */<br>while (true) {<br>wait(R);<br>putc('D');<br>} |
|---|---|---|

Please answer the following questions. (Hint: Process 2 or Process 3 has to wait for Process 1 to run at least once with semaphore R initialized to 0).

1) What is the minimum and the maximum number of D's that might be printed when this set of processes runs? (Hint: If a fixed number of D will be printed, the minimum and the maximum are the same)
2) What is the minumum and the maximum number of A's that might be printed when this set of processes runs? (Hint: If a fixed number of A will be printed, the minimum and the maximum are the same)
3) Is CABDCDABABCABDCD a possible output sequence when this set of processes runs?
4) Is CABABACDBCABDDCD a possible output sequence when this set of processes runs?

1. Maximum and minimum is 4.
2. Smallest 0 and maximum infinity. This is because the process 2 could go on to rerun infinity times because it waits and signals R
3. No
4. Yes

**3. [30 points] Deadlocks**

Consider the following snapshot of a system:

·TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

| | Allocation | | | | | Max | | | | | Available | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **A** | **B** | **C** | **D** | | **A** | **B** | **C** | **D** | | **A** | **B** | **C** | **D** |
| **P0** | 0 | 1 | 0 | 0 | | 0 | 1 | 1 | 3 | | 2 | 2 | 0 | 1 |
| **P1** | 0 | 0 | 0 | 2 | | 1 | 0 | 1 | 2 | | | | | |
| **P2** | 1 | 0 | 1 | 0 | | 3 | 7 | 1 | 3 | | | | | |
| **P3** | 0 | 1 | 0 | 0 | | 2 | 1 | 0 | 0 | | | | | |
| **P4** | 0 | 3 | 1 | 0 | | 1 | 6 | 1 | 1 | | | | | |

Answer the following questions using the banker's algorithm:

1) (5 points) What is the content of the matrix *Need* denoting the number of resources needed by each process?

| | Need | | | |
|---|---|---|---|---|
| | **A** | **B** | **C** | **D** |
| **P0** | 0 | 0 | 1 | 3 |
| **P1** | 1 | 0 | 1 | 0 |
| **P2** | 2 | 7 | 0 | 3 |
| **P3** | 2 | 0 | 0 | 0 |
| **P4** | 1 | 3 | 0 | 1 |

2) (10 points) Is the system in a safe state? Why?

**Yes**, (P3, P4, P1, P0, P2) shows that the system is in a safe state.

Init state:

| | | | |
|---|---|---|---|
| 2 | 2 | 0 | 1 |

| | Allocation | | | | | Max | | | | Order | Available | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **A** | **B** | **C** | **D** | | **A** | **B** | **C** | **D** | | **A** | **B** | **C** | **D** |
| **P0** | 0 | 1 | 0 | 0 | | 0 | 1 | 1 | 3 | 4 | 2 | 7 | 1 | 3 |
| **P1** | 0 | 0 | 0 | 2 | | 1 | 0 | 1 | 2 | 3 | 2 | 6 | 1 | 3 |
| **P2** | 1 | 0 | 1 | 0 | | 3 | 7 | 1 | 3 | 5 | 3 | 7 | 2 | 3 |

·TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **P3** | 0 | 1 | 0 | 0 | | 2 | 1 | 0 | 0 | 1 | 2 | 3 | 0 | 1 |
| **P4** | 0 | 3 | 1 | 0 | | 1 | 6 | 1 | 1 | 2 | 2 | 6 | 1 | 1 |

**Answer:** Yes, (P3, P4, P1, P0, P2)

3) (5 points) If a request from process P0 arrives for (0, 1, 0, 1), can the request be granted immediately? Why?

No, because it doesn't fulfil the first requirement that $Request \leq Need$ which in this case is $(0,1,0,1) \leq (0,0,1,3)$.

4) (10 points) If a request from process P2 arrives for (0, 1, 0, 1), can the requested be granted immediately? Why?

No. The request fulfils the first two requirements where $Request \leq Need, Request \leq Available$ but the system is not in a safe state and therefore can't process the request immediately.

**4. [30 points] Memory Management**

1) (15 points) Part of the page table is shown here:

| page | frame |
|---|---|
| 0 | 5 |
| 1 | 4 |
| 2 | 3 |
| 3 | 2 |
| 4 | 7 |

What are the physical addresses for the following logical addresses in binary?
a. 00111110
b. 11000101
c. 10101110
d. 01010101
e. 01101011

Because the logical address is 8 $bits$ $2^m = 8 \rightarrow\ m = 3$
A) 001 = Page 1. $Physical\ address = (3 * 4) + 30 = 42$
B) 110 = Page 6. Page 6 is not shown
C) 101 = Page 5. Page 5 is not shown
D) 010 = Page 2. $Physical\ address = (3 * 3) + 21 = 30$
E) 011 = Page 3. $Physical\ address = (2 * 3) + 11 = 17$

·TA responsible for HW3: (Wenxue Li, wlicv@connect.ust.hk)

2) (15 points) Consider the following segment table:

| Segment | Base | Length |
|---|---|---|
| 0 | 219 | 600 |
| 1 | 2300 | 14 |
| 2 | 90 | 100 |
| 3 | 1327 | 580 |
| 4 | 1952 | 96 |

What are the physical addresses for the following logical addresses?
a. 0,410
b. 1,10
c. 2,500
d. 3,380
e. 4,112

A) 410+219=629
B) 10+2300=2310
C) Illegal address because the length of segment 2 is 100 and offset is 500. ($100 < 500$)
D) 380+1327=1707
E) Illegal address because the length of segment 4 is 96 and offset is 112. ($96 < 112$)