FALL 2017 COMP 3511 Homework Assignment #3 Solution

Handout Date: October 30, 2017 Due Date: November 11, 2017

Name: ______________ ID: ___________________E-Mail: _____________

Please read the following instructions carefully before answering the questions:

- You should finish the homework assignment **individually**.
- There are a total of **4** questions.
- When you write your answers, please try to be precise and concise.
- Fill in your name, student ID, email and Section number at the top of each page.
- Please fill in your answers in the space provided, or you can type your answers in the MS Word file.

- **Homework Collection: the hardcopy** is required and the homework is collected in collection box #20. The collection boxes locate outside Room 4210, near lift 21 (there are labels attached on the boxes).

1. (20 points) Multiple choices

   1) All processes share a semaphore variable mutex, initialized to 1. Each process must execute wait(mutex) before entering the critical section and signal(mutex) afterward.

      i) Suppose a process executes in the following manner :
      signal(mutex);
      …..
      critical section
      …..
      wait(mutex);

      In this situation :
      A) a deadlock will occur
      B) processes will starve to enter critical section
      C) several processes maybe executing in their critical section
      D) All of these

      Answer: C

   2) A deadlock free solution to the dining philosophers problem :
      A) necessarily eliminates the possibility of starvation
      B) does not necessarily eliminate the possibility of starvation
      C) eliminates any possibility of any kind of problem further
      D) None of these

      Answer: B

3) A system is in the safe state if
A) the system can allocate resources to each process in some order and still avoid a deadlock
B) there exist a safe sequence
C) both (a) and (b)
D) none of the mentioned

Answer: C

4) To ensure no preemption, if a process is holding some resources and requests another resource that cannot be immediately allocated to it :
A) then the process waits for the resources be allocated to it
B) the process keeps sending requests until the resource is allocated to it
C) the process resumes execution without the resource being allocated to it
D) then all resources currently being held are preempted

Answer: D

5) One way to ensure that the circular wait condition never holds is to ________.

A) impose a total ordering of all resource types and to determine whether one precedes another in the ordering
B) to let a process wait for only one resource at a time
C) to never let a process acquire resources that are held by other processes
D) All of these.

Answer: A

6) A system with 5 processes P0 through P4 and three resource types A, B, C has A with 10 instances, B with 5 instances, and C with 7 instances. At time t0, the following snapshot has been taken :

Process
P0
P1
P2
P3
P4

Allocation (process-wise : P0 through P4 top to bottom)
A B C
0 1 0
2 0 0
3 0 2

2 1 1
0 0 2

Max (process-wise : P0 through P4 top to bottom)
A B C
7 5 3
3 2 2
9 0 2
2 2 2
4 3 3

Available
A B C
3 3 2
The sequence leads the system to :
A) an unsafe state
B) a safe state
C) a protected state
D) a deadlock

Answer: B

7) If we preempt a resource from a process, the process cannot continue with its normal execution and it must be :
A) aborted
B) rolled back
D) terminated
D) queued

Answer: B

8) Run time mapping from virtual to physical address is done by
A) memory management unit
B) CPU
C) PCI
D) none of the mentioned

Answer: A

9) If there are 32 segments, each of size 1Kb, then the logical address should have :

A) 13 bits

B) 14 bits

C) 15 bits

D) 16 bits

Answer: C

10) Consider a computer with 8 Mbytes of main memory and a 128 K cache. The cache block size is 4 K. It uses a direct mapping scheme for cache management. How many different main memory blocks can map onto a given physical cache block ?
A) 2048
B) 256
C) 64
D) 8

Answer: C

2. (30 points) Please answer the following questions in a few sentences

1) (6 points) List the factors on the basis of which the process to be aborted is chosen.

Answer: priority of the process; process is interactive or batch; how long the process has computed; how much more long before its completion; how many more resources the process needs before its completion; how many and what type of resources the process has used (one points each)

2) (5 points) Discuss the tradeoff in terms of fairness and throughput in the solution to the first readers-writers problem. Propose a method for solving the readers- writers problem without causing starvation.

Answer: Throughput in the readers-writers problem is increased by favoring multiple readers as opposed to allowing a single writer to exclusively access the shared values. On the other hand, favoring readers could result in starvation for writers. The starvation in the readers- writers problem could be avoided by keeping timestamps associated with waiting processes. When a writer is finished with its task, it would wake up the process that has been waiting for the longest duration. When a reader arrives and notices that another reader is accessing the database, then it would enter the critical section only if there are no waiting writers. These restrictions would guarantee fairness.

3) (4 points) Compare the circular-wait scheme used in deadlock prevention with the deadlock-avoidance schemes (like the banker's algorithm) with respect to the following two issues:

   a. Runtime overheads

   b. System throughput

Answer: A deadlock-avoidance scheme tends to increase the runtime overheads due to the cost of keep track of the current resource allocation. However, a deadlock-avoidance scheme allows for more concurrent use of resources than schemes that statically prevent the formation of dead- lock. In that sense, a deadlock-avoidance scheme could increase system throughput.

4) (5 points) Consider a system consisting of four resources of the same type that are shared by three processes, each of which needs at most two resources. Show that the system is deadlock-free.

   Answer: Suppose the system is deadlocked. This implies that each process is holding one resource and is waiting for one more. Since there are three processes and four resources, one process must be able to obtain two resources. This process requires no more resources and, therefore it will return its resources when done.

5) (5 points) Given five memory partitions of 100 KB, 500 KB, 200 KB, 300 KB, and 600 KB (in order), how would each of the first-fit, best-fit, and worst-fit algorithms place processes of 212 KB, 417 KB, 112 KB, and 426 KB (in order)? Which algorithm makes the most efficient use of memory?

Answer:
First-fit: 212K is put in 500K partition; 417K is put in 600K partition; 112K is put in 288K partition (new partition 288K = 500K − 212K); 426K must wait.
Best-fit: 212K is put in 300K partition; 417K is put in 500K partition; 112K is put in 200K partition; 426K is put in 600K partition.
Worst-fit: 212K is put in 600K partition; 417K is put in 500K partition; 112K is put in 388K partition; 426K must wait.
In this example, best-fit turns out to be the best.

6) (5 points) Explain the difference between internal and external fragmentation.

Answer: Internal Fragmentation is the area in a region or a page that is not used by the job occupying that region or page. This space is unavailable for use by the system until that job is finished and the page or region is released.

3. (35 points) Consider the following snapshot of a system:

1)

	Allocation	Max	Available
	A B C D E	A B C D E	A B C D E
P_0	1 0 0 1 0	4 2 1 2 3	1 4 2 1 1
P_1	2 1 0 1 1	2 3 1 6 1	
P_2	1 0 1 2 2	1 4 2 3 2	
P_3	1 1 3 2 0	3 6 6 8 4	
P_4	3 1 2 1 0	5 2 5 2 2	

Answer the following questions using the banker's algorithm:

a. (8 points) Illustrate that the system is in a safe state by demonstrating an order in which the processes may complete.

Answer: P2, P4, P0, P1, P3.

b. (6 points) If a request from process P0 arrives for (0, 1, 0, 1), can the request be granted immediately?

Answer: No. The need matrix is as follows:

Need
A B C D E
3 2 1 1 3
0 2 1 5 0
0 4 1 1 0
2 5 3 6 4
2 1 3 1 2

Since request0 < need0, update available,

Allocation	Need	Available

	A	B	C	D	E		A	B	C	D	E		A	B	C	D	E
P0	1	0	0	1	0		3	2	1	1	3		1	3	2	0	1
P1	2	2	0	2	1		0	1	0	4	0						
P2	1	0	1	2	2		0	4	1	1	0						
P3	1	1	3	2	0		2	5	3	6	4						
P4	3	1	2	1	0		2	1	3	1	2						

The current state is NOT safe, since available resource D is less than the need of every process.

a. (6 points) If a request from process P4 arrives for (1, 0, 1, 0), can the request be granted immediately?

Answer: Yes. The need matrix is as follows:

Need				
A	B	C	D	E
2	2	0	1	2
0	2	1	5	0
0	4	1	1	0
2	5	3	6	4
2	1	3	1	2

Since request4 < need4, update available,

	Allocation						Need						Available				
	A	B	C	D	E		A	B	C	D	E		A	B	C	D	E
P0	1	0	0	1	1		3	2	1	1	2		1	4	2	1	0
P1	2	1	0	1	1		0	2	0	5	0						
P2	1	0	1	2	2		0	4	1	1	0						
P3	1	1	3	2	0		2	5	3	6	4						
P4	3	1	2	1	0		2	1	3	1	2						

The current state is safe. A feasible execution sequence is P2, P4, P0, P1, P3.

2) (15 points) Consider a system consisting of m resources of the same type, being shared by n processes. Resources can be requested and released by processes only one at a time. Show that the system is deadlock free if the following two conditions hold:

a. The maximum need of each process is between 1 and m resources

b. The sum of all maximum needs is less than $m + n$

Answer: Using the terminology of Section 7.6.2, we have:

a. $\sum_{i=1}^{n} Max_i < m + n$

b. $Max_i \geq 1, for\ all\ i$

Proof: $Need_i = Max_i - Allocation_i$

If there exists a deadlock state, then:

b. $\sum_{i=1}^{n} Allocation_i = m$

Use a. to get: $Need_i + Allocation_i = Max_i < m + n$

Use c. to get: $Need_i + m < m + n$

Rewrite to get: $\sum_{i=1}^{n} Need_i < n$

This implies that there exists a process P_i such that $P_i = 0$. Since $Max_i \geq 1$, it follows that P_i has at least one resource that it can release. Hence the system cannot be in a deadlock state.

c. (15 points) Memory management

Consider the following segment table:

Segment	Base	Length
0	220	540
1	2400	600
2	100	90
3	150	60
4	800	200

What are the physical addresses for the following logical addresses?

a. 0,300 b. 1,100 c. 2,500 d. 3,40 e. 4,250

Answer:

a. 220+300=520

b. 2400+100=2500

c. illegal reference, trap to operating system

d. 150 + 40 = 190

e. illegal reference, trap to operating system

(3 points each)