ANSWER
Do NOT PRINT

NATIONAL UNIVERSITY OF SINGAPORE

CS2106 – INTRODUCTION TO OPERATING SYSTEMS

(Semester 1: AY2016/17)

ANSWER BOOKLET

Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This answer booklet consists of **SIX (6)** printed pages.
2. Fill in your Student Number clearly on all odd-numbered pages.

STUDENT NUMBER (fill in with a pen):

A								

For examiner's use only		
Question	***Total***	***Marks***
Q1-8	16	
Q9	9	
Q10	14	
Q11	16	
Q12	14	
Q13	9	
Q14	22	
TOTAL	**100**	

Write your answers for the MCQs in the boxes below.

1.	B	2.	A	3.	D	4.	E
5.	C	6.	A	7.	E	8.	E

Write your answers in the box/space provided.

9. [9]

Process Management

Abstraction: **illusion that process executes on CPU all the time.**

Protection: **Execution context of each process is isolated from each other.**

Memory Management

Abstraction: **illusion that process owns the entire memory space.**

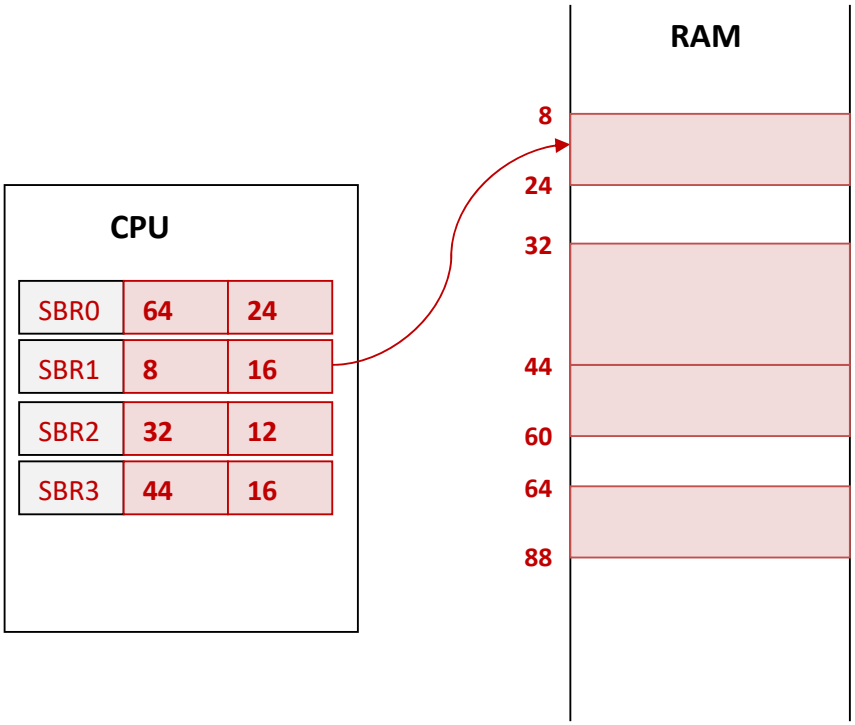
Protection: **Memory space of each process are mapped to different physical address, isolating them from each other.**

File Management

Abstraction: **Files is a single contiguous logical entity.**

Protection: **Files can only be opened through system call, OS can prevent files from being opened for incompatible operations.**

Marks for Q9: /9

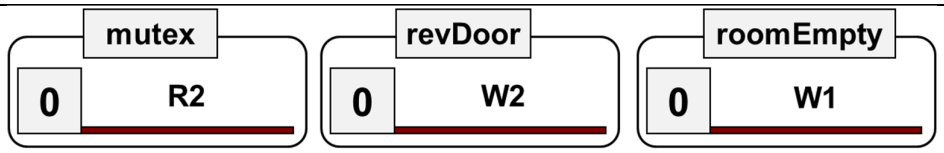
10a. [4]	
10b. [3]	As the heap region is already bordering on the text region. We need to find a larger free piece of memory for relocating the heap region. SBR2 will store new base and limit afterwards.
10c. [3]	We ~~need~~ / do not need TLB in this machine The SBRs are hardware registers, which give very fast access to the physical address lookup already.
10d. [4]	Key ideas: - Only stack region needs to be switched. So, the SBR0 should be saved as part of a text context. - Upon switching, we simply swap the SBR0 to the one saved in the thread context.

Marks for Q10: /14

11. [4x4]

The snapshot is legal / ~~illegal~~.

R1 to R3 reading data.

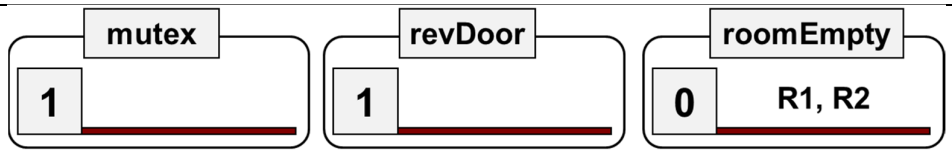


The snapshot is legal /~~illegal~~.

W1 blocks on roomEmpty ➔ readers in the room.

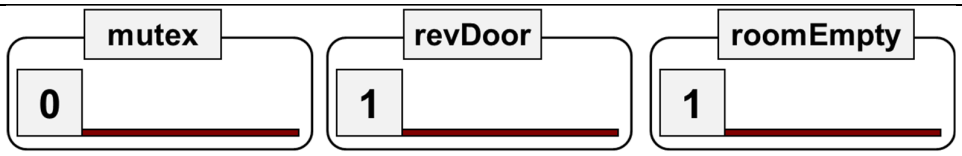
R2 blocks on mutex ➔ another reader is in CS.

Hence, R1 reading data, R3 in CS is one correct answer.



The snapshot is ~~legal~~ / illegal.

nReader update is "atomic" ➔ no way for two readers to find nReader == 1 and blocks on roomEmpty.



The snapshot is legal / ~~illegal~~.

Many answers. Simplest = the first reader is in the CS, before wait(roomEmpty). All other tasks DoingSomethingA/B

Marks for Q11: /16

12a. [2]	Criterion 1 Requests are in the same or nearby sector (can consider cluster size). Criterion 2 Requests are of the same type, read / write.
12b. [2]	Advantage: Seeking latency is reduced.
12c. [4]	Disadvantage: Potential starvation for user process if the request is not near to existing requests. Mitigate: Take the request time into account and set certain deadline. Once the deadline is near, issue request regardless of whether it can be merged.
12d. [3]	Reason to delay: Disk I/O request has very high latency. Delaying the user request will not increase the time very much. However, with more user requests pending, OS can optimize the I/O better. For example, if we do not have enough I/O requests to choose from, merging will not be very effective.
12e. [3]	Potential conflict: It may turn out that the harddisk controller schedule the requests differently. In the worst case, the scheduling decision by OS may be undone by the controller ➔ time used for sorting / merging are wasted.

Marks for Q12: /14

13a. [2]	Swap ~~should~~ / should not be handled as a normal file. Reason: Normal file may be spread across different locations on the secondary storage. Paging performance will be affected.
13b. [4]	Relationship: Page size should be the same or multiple of cluster size. Reason: Pages can be efficiently swapped out.
13c. [3]	Reason: As in (a), OS can preallocate a continuous stretch in secondary storage for the system wide swap file. It is also hard to predict the memory usage of a user program.

14a. [4]	Directory Content: Files = FF.txt, GG.txt, N.txt. Subdirectory = P.txt
14b. [4]	Absolute file path for "N.txt" = Root / Q.txt / N.txt Absolute file path for "V.txt" = Root / Q.txt / P.txt / V.txt
14c. [4]	Disk blocks for "N.txt" = 15, 3, 12 Disk blocks for "V.txt" = 0
14d. [3]	Affected entries: FAT: 4 FREE; Disk Block: 7 R.txt 0 15
14e. [3]	Starting disk block number = 11
14f. [4]	Affected entries: FAT: 5 15; Disk Block: 6 N.txt 0 5

~~~ End of Paper ~~~

Marks for Q13: /9

Marks for Q14: /22
~~~