**Answer**

**Do NOT Print!!**



**NATIONAL UNIVERSITY OF SINGAPORE**

# CS2106 – INTRODUCTION TO OPERATING SYSTEMS

(Semester 1: AY2018/19)

# 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 | | |
|---|---|---|
| ***Questions*** | ***Total*** | ***Marks*** |
| Q1-6 (page 2) | 24 | |
| Q7-11 (page 3) | 22 | |
| Q12-17 (page 4) | 22 | |
| Q18-22 (page 5) | 24 | |
| Q23-25 (page 6) | 8 | |
| **TOTAL** | **100** | |

**1.** [4]

Semaphore P(1), Q(0)

| Task A: | Task B: |
|---|---|
| wait ( P) | wait ( Q) |
| …. | …. |
| signal(Q) | signal(P) |

**2.** [4]

Independence: Task B can be blocked if it reaches the critical section first even though A is not anywhere near.

Progress: Similarly, as there is no task in critical section, Task B should not be blocked.

**3.** [4]

Yes, when the memory load cause a page fault. Swap pages need to be brought in, i.e. disk I/O.

**4.** [4]

Yes, OS / Library can have in-memory buffer for file content to provide. So, the file operation actually just read from the buffer instead of the file, i.e. no disk I/O.

**5.** [4]

The page directory only.

**6.** [4]

Cannot rename the file.

Cannot delete the file.

**7.** [4]

A < B.

Internal fragmentation. A file may not fully occupy the last logical block.

**8.** [4]

Yes, reduce fragmentation.

**9.** [5]

Semaphore Declaration(s):

Semaphore M(1)

| | | | | | |
|---|---|---|---|---|---|
| A1 | wait( M ) | B1 | NA | C1 | wait( M ) |
| A2 | signal( M ) | B2 | NA | C2 | signal( M ) |

**10.** [5]

| | | | | | |
|---|---|---|---|---|---|
| **A** | **Low / High** | **B** | **Medium** | **C** | **High / Low** |

A (low priority) lock M and blocks C. B get to run.

**11.** [4]

Task should release semaphores before blocking.

| | |
|---|---|
| **12.** [2] | **Memory frame replaced is __frame 2___** |
| **13.** [5] | 1. Search through all process's PTE<br>a. Find one with frame 2<br>b. Update to non-memory resident<br>2. Use current process's page table<br>a. update page 8 to be memory resident and in frame 2 |

| | |
|---|---|
| **14.** [2] | **Memory frame replaced is ___frame 0_____** |
| **15.** [4] | |

| | | |
|---|---|---|
| **0** | **B** | **13** |
| **1** | **A** | **31** |
| **2** | **A** | **08** |
| **3** | **A** | **17** |

| | |
|---|---|
| **16.** [5] | 1. Use Inverted table at index 2<br>a. Locate affected PTE, change to non-memory resident<br>2. Use current process's page table<br>a. update page 8 to be memory resident and in frame 2 |
| **17.** [4] | **The replacement algorithm is ___Global__, because** memory pages are kept in one single chain in OS ➔ a process can kick out another process's page. |

**18.** [4]

**They are essentially the same, assuming that the processes get to run fairly evenly. Only recently used pages are in the memory frame ➔ working set of process are in the memory.**

**19.** [5]

| | +00 | +01 | +02 | +03 | +04 | +05 | +06 | +07 | +08 | +09 |
|---|---|---|---|---|---|---|---|---|---|---|
| 00 | | | | FR | | | | | | |
| 10 | | | | | | | FR | | | |

**Data Block modified = _____15_____**

**Directory Entry modified = [E5]HATE | 0 | 3 | 1234**

**20.** [5]

| | +00 | +01 | +02 | +03 | +04 | +05 | +06 | +07 | +08 | +09 |
|---|---|---|---|---|---|---|---|---|---|---|
| 00 | | | | | | | | | | |
| 10 | | | | | | | | | | |

**Data Block modified = ____15_____**

**Directory Entry modified = ILOVE | 0 | 3 | 1234**

**21.** [5]

| | +00 | +01 | +02 | +03 | +04 | +05 | +06 | +07 | +08 | +09 |
|---|---|---|---|---|---|---|---|---|---|---|
| 00 | | | | | | | | END | | |
| 10 | | | FR | | | | | | | |

**Data Block modified = _13, 18, 9, 2, 7, 15_____**

**Directory Entry modified = TIS | 0 | 13 | 4333**

**22.** [5]

| | +00 | +01 | +02 | +03 | +04 | +05 | +06 | +07 | +08 | +09 |
|---|---|---|---|---|---|---|---|---|---|---|
| 00 | | END | | | | | | | | |
| 10 | | | | | 01 | | | | | |

**Data Block modified = ___14, 1, 5_____**

**Directory Entry modified = TIS | 0 | 13 | 5567**

| | |
|---|---|
| **23.** [2] | **Number of '1's is __15___** |
| **24.** [2] | **Number of '1's is ___5 (2 folders + 3 files)____** |
| **25.** [4] | **Hardest to reach _____2/3/4th Block of "/WHY/FAT08"_____**<br>**Number of disk accesses = ___7 = 1 ( "/" inode) + 1("/" DEs) + 1( "WHY/" inode) + 1 ("WHY/" DEs) + 1 ("FAT08" inode) + 1 (single indirect) + 1 (file content)____** |