Name ______________________

Student Number ______________________

Instructor: S. Qiao

SFWR ENG/COMP SCI 3SH3

Day Class
Duration of examination: two hours
McMaster University Final Examination

April 2013

This examination paper includes **6** pages and **12** questions. You are responsible for ensuring that your copy of the paper is complete. Bring any discrepancy to the attention of your invigilator.

SPECIAL INSTRUCTIONS: This paper must be returned with your answers. Open book and notes, no electronics, use of McMaster standard (Casio-FX991) calculator is allowed.

1. (2 marks) Which of the following statements is *false*? If you think all are true, you may answer none.

(a) Virtual memory allows the degree of multiprogramming to be raised.
(b) Virtual memory allows all processes to be run faster.
(c) Virtual memory allows large processes to be run.

Answer: (b)

2. (2 marks) In a demand-paging system where TLB (translation look-aside buffer) is used, which of the following statements is true? If you think all are false, you may answer none.

(a) A TLB miss is always followed by a page fault.
(b) A page fault is always preceded by a TLB miss.
(c) A TLB hit is always followed by a page fault.

Answer: (b)

3. (2 marks) Which of the following memory organizations eliminates external fragmentation?

(a) pure paging
(b) segmentation
(c) paged segmentation

Answer: (a)

next page ....

4. (2 marks) Given the memory size in a pure paging system, a small page size implies

(a) small page table entry size
(b) large page table entry size
(c) none of the above (no effect on page table entry size)

Which of the above statements is true?

Answer: (b)

5. (2 marks) Given the memory size in a pure paging system, a small page size implies

(a) more external fragmentation
(b) more internal fragmentation
(c) less external fragmentation
(d) less internal fragmentation

Which of the above statements is true?

Answer: (d)

6. (6 marks) Assuming a pure paging system on a 16-bit (both virtual and physical address spaces) machine with page size of 256 bytes and the following page table of size 4:

valid	used	dirty	readOnly	virtual page number	physical page number
T	T	F	T	00	0A
T	T	F	T	01	0B
F	F	F	F	02	00
T	T	T	F	03	12

Translate the following virtual addresses into physical addresses. All numbers are hexadecimal.

(a) 000A read

virtual page number = 00 → physical page number 0A
physical addr. 0A0A

(b) 0123 write

virtual page number: 01
read only violation

(c) 0234 write

virtual page number: 02
invalid physical page, page fault

(d) 0456 read

bound violation

7. (2 marks) A computer has six tape drives, with n processes competing for them. Each process needs at most two drives. What is the maximum value of n which guarantees the system deadlock free?

(a) Three
(b) Four
(c) Five

Answer: (c)

8. (4 marks) A system that uses Banker's Algorithm for checking safe state has five processes (1, 2, 3, 4, and 5) and uses resources of four different types (A, B, C, and D). Resource type A has total of 13 instances, B 13, C 9, and D 13. The following is a snapshot of a system. Is the state of the system safe? If the system is safe, show a safe sequence. If the system is unsafe, show how deadlock might occur.

Max

	A	B	C	D
1	4	2	2	0
2	3	7	2	3
3	4	11	9	8
4	10	5	5	11
5	8	4	2	6

Allocation

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

where Max gives the maximum needs of the processes and Allocation gives the current resource allocations of the processes.

Answer:
The matrix Need is

Need

	A	B	C	D
1	3	2	0	0
2	3	4	1	1
3	2	7	4	7
4	7	5	5	5
5	4	2	1	3

The vector of available resources is

(13 13 9 13) − (10 9 9 12) = (3 4 0 1)

Safe sequence or deadlock?

	3	4	0	1
p1	1	0	2	0
	4	4	2	1
p2	0	3	1	2
	4	7	3	3
P5	4	2	1	3
	8	9	4	6

deadlock

next page ....

9. (2 marks) Typically, execution of a program is a sequence of

(a) CPU burst, I/O, CPU burst, I/O, ..., I/O.
(b) I/O, CPU burst.
(c) CPU burst, I/O, CPU burst, I/O, ..., CPU burst.

Answer: (c)

10. A CPU scheduling uses priority algorithm. Each process is given a base priority between -20 to $+20$. A smaller number represents a higher priority. In t seconds, the scheduling priority is calculated by adding the adjustment

$$\text{adj} = \text{recent CPU usage} \times r^t$$

to the base priority. The decaying rate r is obtained from

$$0.1 = r^{5n}, \quad \text{that is} \quad r = 10^{-1/(5n)},$$

where n is the average number of ready processes in the last minute.

(i) (2 marks) When a process changes from CPU bound to I/O bound, its scheduling priority gets

(a) higher
(b) lower
(c) unchanged

Answer: (a)

(ii) (2 marks) Which of the following statements is *false*? If you think all are true, you may answer none.

(a) This scheduling method favors CPU bound processes.
(b) This scheduling method favors I/O bound processes.
(c) This scheduling method is adaptive.

Answer: (a)

next page ....

11. (6 marks) In Nachos-4.02, to implement `WaitUntil`, we introduce a structure PendingAlarm

- wake-up time
- alarmSemaphore with initial value 0

and a sorted list `pendingAlarmList`. When `Alarm` is constructed, `pendingAlarmList` and a lock are constructed. The following is an outline of the implementation of `WaitUntil`:

```
Alarm::WaitUntil

    Calculate wake-up time;
    Construct a pendingAlarm using the wake-up time,
      which constructs alarmSemaphore;
    Acquire the lock for pendingAlarmList;
    Insert the pendingAlarm to the list;
    Release the lock
    Call alarmSemaphore->P();
    Delete pendingAlarm;
```

Modify `Alarm::CallBack` as part of the implementation of `WaitUntil`.

Answer:

```
Alarm::CallBack

    Get machine status;
```

while pendingAlarmList not empty and
its front pending alarm is due

Remove the front alarm from the list

Call alarmSemaphore -> V( )

```
    if status is not idleMode
       Call interrupt->YieldOnReturn;
```

12. (6 marks) In a file system, a file header (inode, in UNIX term) contains ten pointers (integers) for data sectors. Among the ten pointers, nine are pointers to direct data sectors and one is the pointer to double indirect sectors. Assuming the size of a disk sector is 512 bytes, what is the maximum file size in this system? Note that `sizeof(int) = 4`. Show and explain your calculation.

Answer:

Direct : $9 \times 512 = 4{,}608$

Double indirect

$$\frac{512}{4} \times \frac{512}{4} \times 512$$

$= 8{,}388{,}608$

Max size $8{,}388{,}608 + 4{,}608$

$= 8{,}393{,}216$ bytes

END!