ECE PARIS
ÉCOLE D'INGÉNIEURS

Prénom : ...HaKAYEM...........................
NOM : ....Jérémy.........................
Promotion : 2018
Groupe : ..S111...

52

Inr.

ING 4
Systèmes d'exploitation
Devoir surveillé

16 décembre 2016
14:30 - 16:30
Durée : 02:00

Sujet proposé par :	KHOURY Christian
Calculatrice autorisée :	NON
Documents autorisés :	NON
Ordinateur autorisé :	NON

Vous devez répondre directement sur le sujet sans oublier de noter votre nom, prénom et groupe.

RAPPEL :

- NOM et Prénom de l'élève doivent être portés sur toutes les copies rendues.
- Les copies doivent être numérotées.
- Tous les appareils électroniques (téléphones portables, PDA, ordinateurs, montre connectée, etc.) doivent être éteints et rangés.
- **Toute erreur constatée sur le sujet doit être signalée sur la copie. Le correcteur en tiendra compte lors de la correction du devoir.**
- Il est interdit de communiquer.
- Toute fraude, ou tentative de fraude, qu'elle soit passive ou active, fera l'objet d'un rapport de la part du surveillant et sera sanctionnée par la note zéro, assortie d'une convocation devant le Conseil de discipline. Aucune contestation ne sera possible. Tous les documents et supports utilisés frauduleusement, devront être remis au surveillant.
- Les élèves ne sont pas autorisés à quitter la salle où se déroule l'épreuve moins de 45 minutes après le début de l'épreuve. Au-delà de ces 45 premières minutes, toute sortie est définitive (sauf dans le cas d'une épreuve durant plus de deux heures).

Operating Systems Exam

Documents forbidden

- **Just encircle the most appropriate answer**

1. In the process lifecycle, whenever a process is created, it enters ✓
   a. the RUNNING state
   b. the BLOCKED state
   c. the MOVING state
   [✓ d.] the READY state
   e. none of the above

2. In the process lifecycle, the transistion from the READY state to the RUNNING state indicates that : ✓
   a. a process has been preempted by another process
   b. a process has blocked waiting for an event
   c. a process is done waiting for an event
   d. a process was just created
   [✓ e.] none of the above

3. In the process lifecycle, the transition from the RUNNING state to the BLOCKED state indicates that : ✗
   a. the OS has blocked it for some unknown reason
   [✓ b.] a very slow operation is taking place in the process
   c. a very fast operation is taking place in the process
   d. time has elapsed
   ~~e.~~ none of the above

4. In the process lifecycle, the transition from the RUNNING state to the READY state indicates that : ✓
   [✓ a.] the scheduling algorithm is preemptive — preempt or time-out
   b. the scheduling algorithm is non preemptive
   c. the scheduling algorithm is Round Robin
   d. none of the above

5. In the process lifecycle, the transition from the BLOCKED state to the READY state indicates that : ✗
   a. the OS has unblocked it for some unknown reason
   b. a very slow operation is taking place in the process
   c. a very fast operation is taking place in the process
   [✓ d.] time has elapsed
   [✓ e.] none of the above

File descript tab → File table → Inod table

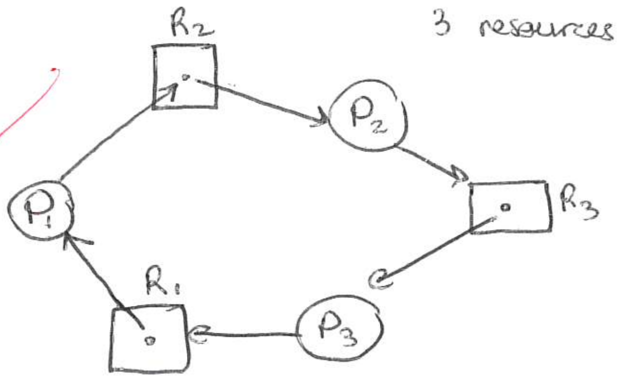
6. Which of the following is shared between all the threads in a process
   a. Register values
   [✓ b.] File descriptor table
   c. Local variables
   d. Stack frames
   e. None of the above

7. Which of the following is not included in an inode ?
   a. File size
   → b. File name
   [✓ c.] File modification date
   d. File permissions
   e. None of the above

inode data struct describe file system
size, horodatage, permiss.

8. Draw a resource allocation graph with 3 processes (1-3) and 3 resources (A-C) where the 3 processes are deadlocked

3 processes : P_1 - P_2 - P_3
3 resources : R_1 - R_2 - R_3



Legend :

R → P :
resource used by process

P → R :
Process waiting for this resource

Given the following 4 tasks (Questions 9 to 14)

	Arrival Time	Exec Time	I/O
T1	2	4	None
T2	0	4	3/1
T3	1	4	3/1
T4	5	4	None

I/O figures (t1/t2) indicate that an I/O call is made at t1 after the start of the process and takes t2 time units to end.

9. Using a FCFS scheduling algorithm, the processes run in the following order :
   → [✓ a.] T2, T3, T1, T2, T4, T3
   b. T2, T3, T1, T4
   c. T2, T3, T2, T1, T4, T3
   d. T3, T1, T2, T4, T2, T3
   e. None of the above

10. Using a FCFS scheduling algorithm, the turnaround time of T3 is

✓

[✓ a.] 15
b. 14
c. 13
d. 12
e. 11
f. None of the above

11. Using a FCFS scheduling algorithm, the waiting time of T3 is

✓

a. 15
b. 14
c. 13
d. 12
[✓ e.] None of the above

12. Using a Round Robin with a quantum of 4, the processes run in the following order

✓

[✓ a.] T2, T3, T1, T2, T4, T3
b. T2, T3, T1, T4
c. T2, T3, T2, T1, T4, T3
d. T3, T1, T2, T4, T2, T3
e. None of the above

13. Using a Round Robin with a quantum of 3, the processes run in the following order

✓

a. T2, T3, T2, T1, T4, T3, T1, T4
[✓ b.] T2, T3, T1, T2, T4, T3, T1, T4
c. T1, T2, T3, T4
d. T3, T2, T1, T4, T3, T1, T4
e. None of the above

14. Using a Round Robin with a quantum of 3, the turnaround time of T3 is

✓

a. 11
b. 12
[✓ c.] 13
d. 14
e. 15
f. None of the above

15. A context switch happens when

✗

[✓ a.] a task is replaced by another one
→ b. a task moves from the READY state to the RUNNING state
→ c. a task moves from the RUNNING state to the Blocked state
→ d. a task is terminated
[✓ e.] all of the above

16. Give a precise definition of semaphores (structure and operations)

A semaphore is a structure containing a queue and a counter used to synchronize the use of a Shared variable by different threads.

2 ✓

Two operations can be executed on a semaphore :

P(semaphore) : when a thread wants to use the variable. Decrements the counter and allows the thread to access to it. x2 /1

V(semaphore) : when a thread finishs to use the variable. Increments the counter. x2 → 1

17. Synchronization is needed when

a. data is shared and read concurrently by different tasks

✓ [✓ b.] data is shared and modified concurrently by different tasks

c. data is not shared and read concurrently by different tasks

d. data is not shared and modified conconurrently by different tasks

e. none of the above

18. Given 3 tasks that increment concurrently a variable i initalized to 44. Possible outcome of these executions is

a. 44, 45, 46, 47

✓ [✓ b.] 45, 46, 47

c. 46, 47

d. All of the above

e. None of the above

19. Write the solution to the previous problem using semaphores (don't forget to initialize the semaphore)

Semaphore mutex initialized to 1.

4 ✓

```
T1 { P(mutex); i++; V(mutex);}.
T2 { P(mutex); i++; V(mutex);}.
T3 { P(mutex); i++; V(mutex); }.
```

20. Given a semaphore intialized to 0, a shared variable i set to 65, and 5 tasks doing « Acquire(S) i++ Release(S) ». The final value of this variable upon completion of all tasks is

✓

- [✓ a.] 65
- b. 66
- c. 67
- d. 68
- e. 69

21. Given a semaphore initialized to 2, a shared variable i set to 65, and 4 tasks doing « Acquire(S) i++ Release(S) ». The final value can be

✓

- a. 66
- b. 67
- c. 68
- d. 69
- [✓ e.] all of the above
- f. none of the above

= 2 bytes

Given a 1-level paging scheme with addresses over 16 bits and a page size of 256 Bytes. (Questions 22 – 28)

2^{16} : total virtual mem size

22. The address can be written as (page number, offset). The number of bits used for the each part is

✓

- a. (10, 7)
- b. (9, 7)
- c. (8, 7)
- [✓ d.] (8, 8)
- e. None of the above

23. The number of entries in the page table is

✓

- a. 64
- b. 128
- [✓ c.] 256
- d. 512
- e. None of the above

add bits

$$\frac{2^{16}}{2^{8}} = 2^{8}$$

page size

24. The size of the page table is

✗

- →? (a.) 512
- b. 1024
- [✓ c.] 2048
- d. 4096
- e. None of the above

$2^{8} \times 2$

Given the following page table

0	1
1	4
2	2
3	8
4	255
5	9

25. The logical address 0x0001 (in hexadecimal) is physically stored in

✓

- a. 0x1001
- [✓ b.] 0x0101
- c. 0x0081
- d. 0x0802
- e. None of the above

$\overbrace{0000\ 0000}.\overbrace{0000\ 0001}$

Frame#1 ⇒ 0000 0001 . 0000 0001

⇒ $2^8 + 1 = 257 = (16^2 + 1)$

en hexa ⇒ 01 01

26. **If the page size is 128 Bytes,** the logical address 0x0001 would be physically in

✓

- a. 0x1001
- b. 0x0101
- [✓ c.] 0x0081
- d. 0x0802
- e. None of the above

$128 \rightarrow \frac{2^{16}}{2^7} = 2^9$

$\underbrace{0000\ 0000 . 0}_{\text{9 bits page}}\underbrace{000\ 0001}_{\text{7 bits for offset}}$

Page #0 offset 1 ⇒ Frame #1 offset 1.

⇒ 0000 0000 . 1000 0001

⇒ $2^7 + 1 = 129$ $(8 \times 16^1 + 1 \times 16^0)$

⇒ en hexa ⇒ 0081

27. The logical address 0x0605 is physically stored in

✓

- a. 0x0805
- b. 0x0905
- c. 0x0205
- d. 0xFF05
- [✓ e.] None of the above

$\underbrace{0000\ 0110}_{\text{Page 6}} . \underbrace{0000\ 0101}_{\text{offset 5}}$

⇒ Frame nb undefined

28. The logical address 0x0307 is physically stored in

- a. 0x0107
- b. 0x0407
- [✓ c.] 0x0807
- d. 0x2008
- [✓ e.] None of the above

$\underbrace{0000\ 0011}_{\text{page 3}} . \underbrace{0000\ 0111}_{\text{offset 7}}$

⇒ Frame nb 8 offset 7

0000 1000 . 0000 0111

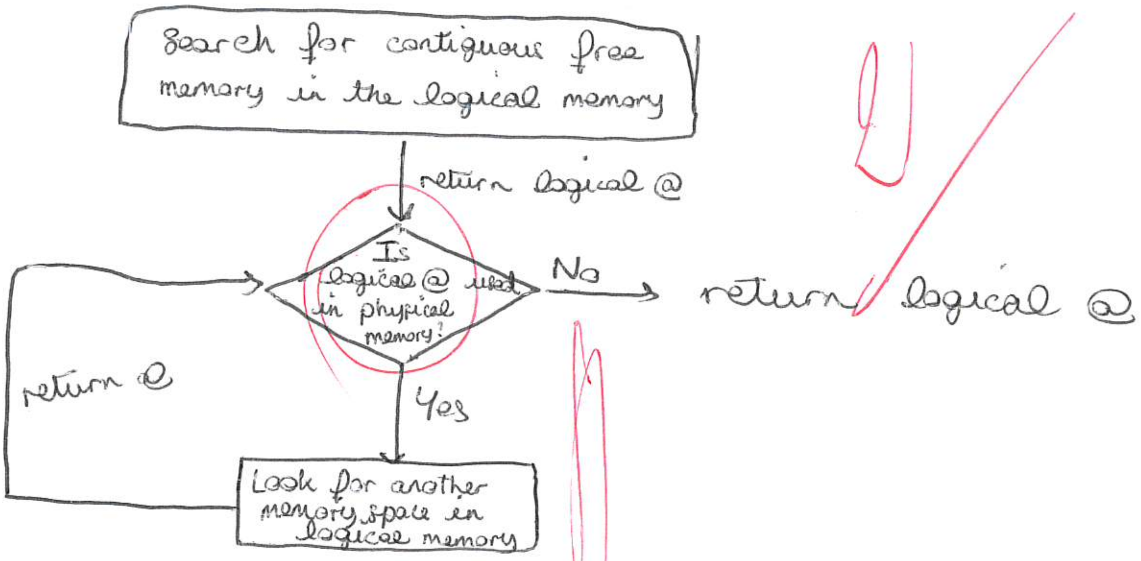
$= 2048 + 7 = 2055$

$= 8 \times 16^2 + 7 = 0801$

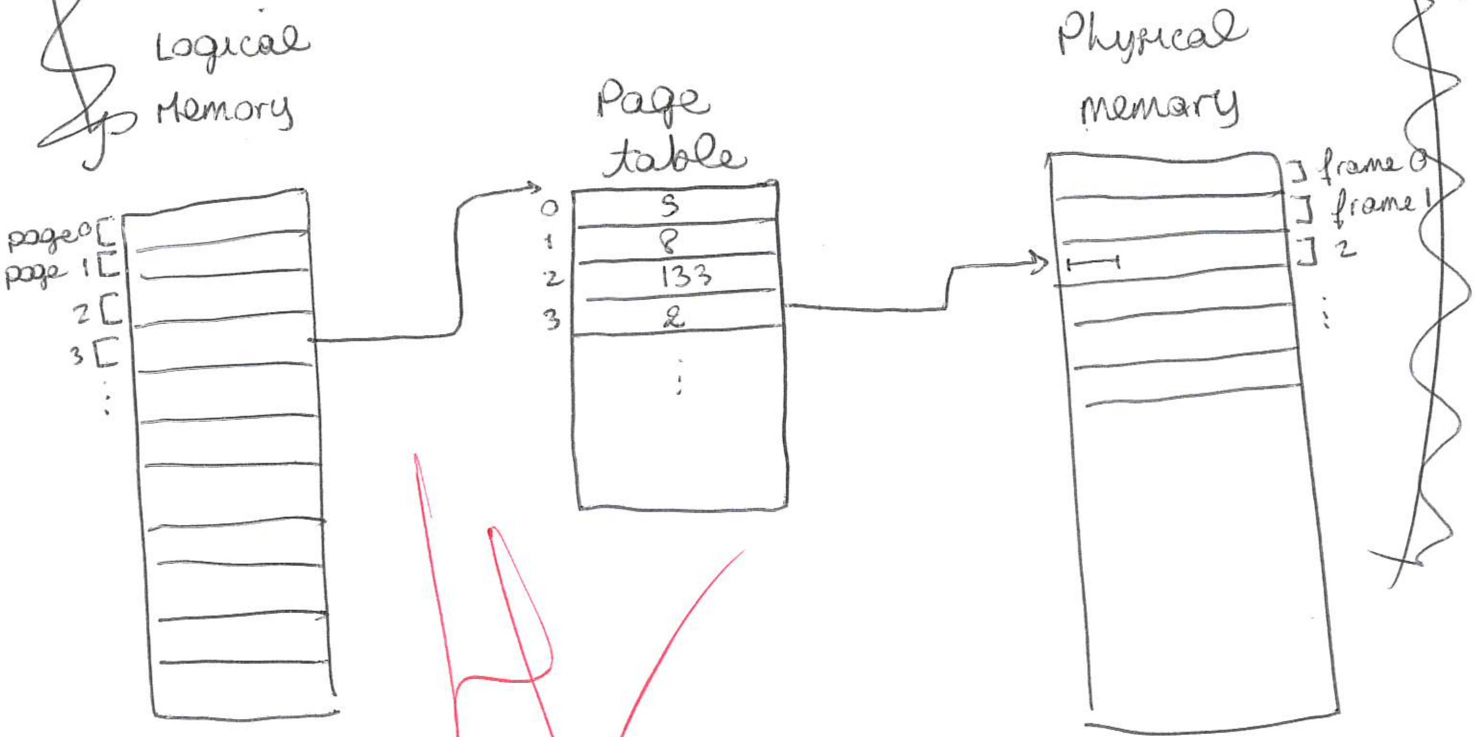
$2^{12} = 4096$

$16^3 = 4096$ $16^2 = 256$

29. Give the algorithm for the allocation in a one-level paging scheme


Search for contiguous free memory in the logical memory
return logical @
Is logical @ used in physical memory?
No
return logical @
Yes
Look for another memory space in logical memory
return @

30. Draw a diagram showing concretely how a logical address is translated into a physical one in a one-level paging scheme using the page table used in the previous questions


Logical Memory
page 0
page 1
2
3
Page table
0
5
1
8
2
133
3
2
Physical memory
frame 0
frame 1
2