COSC 450 Operating System Test #1-1

Name:______________________.

1.

a) Similar job are collected, and save in a magnetic tape and implement one by one sequentially.

b) Jobs for processes (I/O jobs) are saved in a file and executed one by one

c) Since second generation, OS does not support multiprogramming.

d) Process status, snapshot of CPU, scheduling information, memory management information I/O status information.

e)
   a. Protection between jobs
   b. Job scheduling

f) Since instruction cycle are three steps: fetch, encoding, and execute

g) Multiple user are connected to a system and each user are shared system

h) A situation where two or more processes are reading or writing some shared data and the result depends on who runs precisely when, are called race condition.

i) Since process switch need very expensive overhead.

j) Limited size of memory, big size processes.

2.

- 2^{24} = 16 MB
- 2^{20} page tables +1
- there are 2^{20} page table and each has 2^{20} entries. # of pages = $2^{20} \times 2^{20} = 2^{40}$ pages
- size of memory = 2^{32} = 4 GB, there are 4GB /16MB = $4 \times 2^{30} / 16 \times 2^{20} = 2^{32} / 2^{24} = 2^{8}$ page frames. The system need reserve 8 bit for page frame number.

3.

a) $1 - p^n$

b)

n= (8 GB – 2 GB)/1GB = 6

p= 0.9 CPU utilization =1- $(0.9)^6$ = 0.468 about 46%

c) n = (16 GB – 2GB) / 1GB = 14, p = 0.9 1 – $(0.9)^{14}$ = 0.771 about 77 %

4.

If each job has 50% I/O wait, then it will take 40 minutes to complete in the absence of competition. If run sequentially, the second one will finish 80 minutes after the first one starts. With two jobs, the approximate CPU utilization is $1 - 0.5^2$=0.75. Thus, each one gets <u>0.375 CPU minute per minute</u> of real time. To accumulate 20 minutes of CPU time, a job must run for20/0.75 +20/0.75 minutes, or about 53.33 minutes. Thus running sequentially the jobs finish after 80 minutes, but running in parallel they finish after 53.33 minutes.

5.

a) FIFO : 15 page fault

5, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 5, 0, 1

5	0	1	2		3	0	4	2	3	0			1	2			5	0	1
	5	0	1		2	3	0	4	2	3			0	1			2	5	0
		5	0		1	2	3	0	4	2			3	0			1	2	5

b) LRU: 12 page fault

5, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 5, 0, 1

5	0	1	2		2		4	4	4	0			1		1		1		
	5	0	1		3		3	2	2	2			2		2		5		
		5	0		0		0	0	3	3			3		0		0		

c) Optimal: 9 page fault

5, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 5, 0, 1

5	0	1	1		3		3			3			1				1		
	5	0	0		0		4			0			0				0		
		5	2		2		2			2			2				5		

6.

a)

Sol) Assume a short term scheduler select a process with high priority. At time t_0 there is only one process P_L with low priory working on C.S.. At time t_1, process P_H with high priority is in ready state. The scheduler selects P_H for CPU. P_H try to enter C.S. but cannot since P_L is in C.S.. P_H start busy waiting. P_H will busy waiting forever, since P_L never be selected by scheduler.

b)

- For 3KB: page number= 11 offset= 1458
- For 4KB: page number= 8 offset = 2482

7.

a)

virtual address $46851/(4 \times 2^{10}) = 11$ in page #11.
virtual page #11 begin with address $11 \times 4 \times 2^{10}$
page frame #7 is start with address $7 \times 4 \times 2^{10}$
physical address = $7 \times 4 \times 2^{10} + (47891 - 11 \times 4 \times 2^{10}) = 28672 + 2835 = 31507$

b)

virtual address $8765/(4 \times 2^{10}) = 2$ in page #2
virtual page #2 begin with address $2 \times 4 \times 2^{10}$
page frame #1100 is start with address $1100 \times 4 \times 2^{10}$
physical address = $1100 \times 4 \times 2^{10} + (8765 - 2 \times 4 \times 2^{10}) = 4505600 + 573 = 4506173$

8.

- Bitmap: #of allocation unit = 128MB/2KB = $(128 \times 2^{20})/(2 \times 2^{10}) = 2^{27}/2^{11} = 2^{16}$ units
  Size of the bitmap = 2^{16} bits = **2^{13} byte**

- The linked list: number of node for linked list= 128 MB /64KB = $2^{27}/2^{16}$ or 2^{11} nodes.
  size of each node = 32+16+16 = 64 bit = 8 byte $=2^3$ bytes
  Total size of linked list = number of node $\times$ size of a node = $2^{11} \times 2^3$ bytes = **2^{14} bytes**.

COSC 450 Operating System Test #1-2

10/09/2019

Name:________________________.

9.

let's assume Permit =0 at time T0
P0 tries to enter C.S. and can enter since Permit =0.
P0 finish its job in C.S. and set Permit =1
P1 is currently running outside C.S ,it is terminated with fatal error.
P0 tries to enter C.S. again but P0 never can.

10.

let's assume : empty = 0, full = N, mutex =1

- Producer is scheduled : produce item ,down mutex (now mutex =0), try to down empty. Since empty =0, producer cannot finish down operation and sleep on semaphore empty.
- Consumer is scheduled: down full (now full = N-1), then try to down mutex. Since mutex is already down by producer, consumer cannot finish down operation and sleep on semaphore mutex.
- Now producer and consumer sleep forever!

11.

- Long-Term Scheduler – Selects a process from the pool of job and load into memory for execution
- Short-term scheduler – selects a process from the ready queue and allocates the CPU.
- Memory Scheduler – schedule which process is in memory and in disk.

12.

11	8	6	6	6	6	6	6	X
X	9	9	9	9	9	9	9	9
X	X	5	3	3	3	3	X	X
X	X	X	2	1	X	X	X	X
X	X	X	X	2	2	X	X	X

P_1	P_1	P_3	P_4	P_4	P_5	P_3	P_1	P_2

0 3 5 7 8 9 11 14 20 29

- Average waiting time = ((14 – 5) + (20 – 3) + (11- 7) + 0 + (9 -8))/5= 6.2
- Average turnaround time = (20 + (29 – 3) + (14 – 5) + (9 – 7) + (11 – 8))/5 = 12

13.

- First Fit 9(21) -10(12) -15(20)- 18(18)
- Best Fit 9(14) – 10(15)-15(18)-18(20)
- Worst Fit 9 (35) – 10(26) -15 (25) – 18 (23)
- Next Fit 9(21)-10(20)-15(18) -18(25) or
    - 9(21) -10(12)- 15(20) -18(18)

14.

- Sol) virtual address space = $2^{32} = 2^{22} \times 2^{10} = 2^{22}$ **KB**
  $\therefore$ Maximum # of pages per a process = virtual space / a page size = $2^{22}/2 = 2^{21}$ pages .
  Size of page table per a process = number of page $\times$ one entry size
  = $2^{21} \times 64$ bits = $(2^{21} \times 64)/8$ Byte= $2^{21} \times 8$ byte= **2^{24} Byte = 16 MB**

- Sol) simply need calculate the number of page frame
  # of page frame = size of RAM / size of page = 4GB / 2KB = $4 \times 2^{30} / 2 \times 2^{10} = 2^{32}/2^{11} = 2^{21}$ page frames $\therefore$ **21 bits for page frame number.**

15.

1. Lets assume count is currently N, producer read check count == N and then time out before go to sleep.
2. Consumer remove one item from the buffer and reduce count to N-1. The consumer check count is N- 1, call wakeup() to wake up the producer. Since producer did not sleep yet, the signal will be lost.
3. Control back to producer, the producer already checked count =N, so go to sleep.
4. Control back to consumer, consumer will consume all items eventually, then go to sleep. Now, both consumer and producer sleep forever.